A DYNAMIC RANDOM CHANNEL RESERVATION FOR MAC PROTOCOLS IN MULTIMEDIA WIRELESS NETWORKS

A THESIS
SUBMITTED IN PARTIAL FULFILMENT
OF THE REQUIREMENTS FOR THE DEGREE
OF
MASTERS OF SCIENCE IN COMPUTER SCIENCE
IN THE
UNIVERSITY OF CANTERBURY
by
Enoch Chia-chi Kao

University of Canterbury
2001

Examination Committee:

- Supervisor: Associate Prof. Dr. Krzysztof Pawlikowski, University of Canterbury, Christchurch, New Zealand

- Associate supervisor: Associate Prof. Dr. Harsha Sirisena, University of Canterbury, Christchurch, New Zealand

- External examiner: Prof. Dr. Biswanath Mukherjee, University of California in Davis, USA

To my love, Hsin,

for her love, care, and support during the study of this degree. I would not have finished it without her.

Abstract

Medium Access Control (MAC) plays a vital role in wireless networks. With the increasing popularity of multimedia services, MAC protocols of wireless networks are required to satisfy a variety of Quality of Service (QoS) requirements, including short delays. One of techniques for satisfying such requirements is based on assignment of transmission rights on demand. Following such a protocol, bandwidth is assigned to mobile terminals when they have something to transmit. The base station has absolute control of the bandwidth, including assignment of different priorities to different classes of users.

In this thesis, we survey recently proposed MAC protocols for wireless networks. The survey includes MAC protocols designed for different network generations and topologies. Next, we focus on the demand part of demand assignment MAC protocols. We propose a new strategy based on probabilistic assignment that allows mobile terminals to pick the best time for transmitting their demands. Building upon this concept, we propose a new protocol called Transmission Probability Based Dynamic Slot Assignment or, briefly, TRAPDYS. It can be used in existing demand assignment protocols to improve their performance. The TRAPDYS protocol introduces a flexible prioritised access to communication channel by dynamically adjusting transmission rights of mobile terminals to current network traffic activities.

We analyse the performance and behaviour of the TRAPDYS protocol be means of stochastic simulation. The results show that the TRAPDYS protocol is able to cope with a high level of traffic by utilizing temporarily unused network resources and improves the utilization of the demand part of network capacity used by a given demand assignment MAC protocol.

Contents

List of Figures

Chapter 1

Introduction

The increasing popularity of the mobile telephone in recent years has drawn much attention to wireless communication and technologies. With the explosion of the new digital age, wireless communication is no longer simply a radio broadcast or a cordless telephone, but much more. From the digital cellular phone with data capability to the infrared printer port, the wireless technology becomes increasingly important every day. A large leap forward was taken when the second-generation mobile network was released. This is based on digital technologies and is very different from the first-generation mobile network based on analogue technologies. With the emergence of third-generation mobile networks [Pras98], broadband wireless technology is now possible. It provides a bandwidth sufficient for supporting multimedia traffic.

Wireless communication can be based on two types of electromagnetic waves: light waves and radio waves. In the first type, infrared light is used as the transmission medium. As these light waves are not visible to the human eye, they do not interfere with our daily activities. The short wavelengths of infrared light allow high-speed transmission of data, but can be a problem when transmitting over a long distance or when the transmitting target is not in sight. The other common medium is radio waves. The properties of radio waves make them more suitable than light. They can travel a long distance before they are attenuated, and can bend around corners and bounce off walls. Thus, most wireless networks are based on radio waves.

A wireless network consists of wireless nodes. Each node has the ability to transmit and receive radio signals. Similar to the wired network, the wireless network architecture has a layered structure (Figure 1.1), usually containing three or more layers. The lowest layer is the physical layer, which connects to the physical layer of the communicating node with the necessary specifications and arrangements. The transmission methods and technologies are governed by this layer.

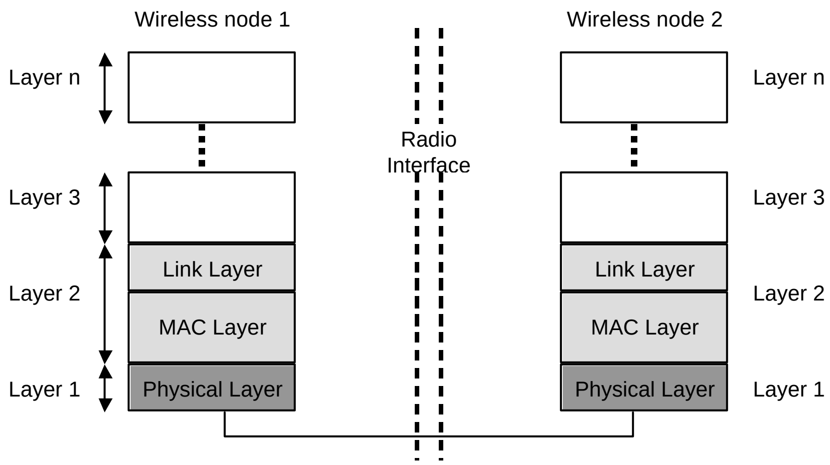

Figure 2.1: Layers of a wireless network architecture.

The second layer is composed of two sub-layers: the medium access control (MAC) layer and the link layer. The MAC layer sits directly on top of the physical layer. The main function of MAC is to control the accessibility of the radio medium. It controls when, how, and who should transmit data. The link layer controls the link between the communicating nodes.

The third layer depends on the type of wireless network. In most, the third layer is the transport layer, and provides a way to exchange data between two nodes.

Due to the propagation properties of a radio frequency, two types of network topologies are available for constructing a wireless network: an ad-hoc topology and a centralized topology. An ad-hoc topology (Figure 1.2a) is also called a distributed topology. It is made up of many wireless nodes, each of which functions identical. There is no specific arrangement between the nodes, and every node in the network is mobile. This topology does not have a defined shape. No central administration is required in the network, as each node is capable of communicating with its neighbour node directly. When a node leaves the network, the network can still function perfectly. Networks based on distributed topologies are called ad-hoc networks.

A centralized topology (Figure 1.2b) is made up of two types of wireless nodes: base station (BS) and mobile terminal (MT). A base station sits in the centre of the topology. Its transmission range defines the size of the system. Since the radio waves are transmitted and travel outwards, the shape of the topology is likely to be a circle. It is commonly described as a cell, similar to a cell in a biological system. The BS serves as the administrator of the system, and is stationary. The MTs are mobile wireless nodes. They communicate only with the BS. All transmissions in a centralized topology must go through the BS. The BS then passes the content of the transmission to the intended receiver. If the BS in a centralized topology fails to function, the entire system fails. A network based on a centralized topology is called a centralized network or a last hop network. It is usually the last hop of a wired network.

Wireless network topology has a great influence on the design of wireless MAC protocols. The details of wireless MAC protocol classification will be discussed in Chapter 3.

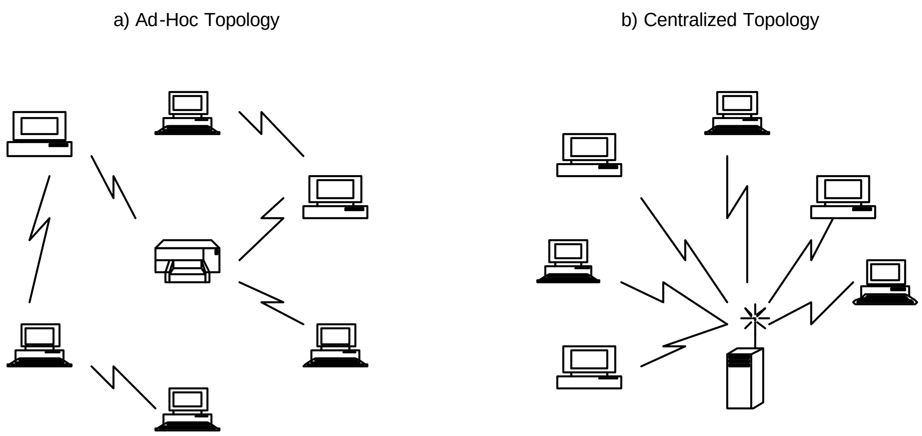

Figure 1.2: Wireless network topologies.

The wireless environment is very different from the wired environment, and hence the design of a MAC protocol must take into account the problems specific to a wireless environment. As radio signals from a node are broadcast, any node that is within its range can hear a given transmission and interference can occur. The emergence of different types of data services, such as multimedia, provides an even greater challenge to MAC protocol design.

A particular type of wireless MAC protocol called a demand assignment MAC protocol has drawn much attention in recent years. These are well suited for third-generation mobile networks. They are designed for a cellular network based on a centralized topology. The MTs send their requests to the BS and wait for the BS to assign bandwidth to them. The demand assignment MAC protocols combine random access and guaranteed access. Random access is a scheme that allows the MTs to access the medium randomly and without many restrictions. Guaranteed access is based on polling. It polls each MT in the network in a round robin fashion. The demand assignment MAC protocols provide the good Quality of Service (QoS) required for complex traffic that exits today.

This research focuses on strategies to improve the performance of existing demand assignment MAC protocols. The MTs use random access when they send their requests to the BS. Such random access provides many benefits that are not found in guaranteed access, including scalability, support of a large MT population, simplicity, and less power usage. However, it is potentially inefficient.

We propose a protocol called transmission probability based dynamic slot assignment (TRAPDYS) to improve the performance of random access in demand assignment MAC protocols. The protocol is based on a new concept called transmission probability assignment. The traffic conditions of the random access part of the demand assignment MAC protocol are observed to enable the MTs to choose time slots that have a high chance of success in transmitting their requests. (The definition of a time slot is described in Chapter 2.) TRAPDYS provides prioritised access for different types of traffic, to utilize the possibly unused bandwidth in different priority classes, and to relieve the traffic loads of the network when bursts occur. The protocol can be implemented as an add-on to some of the existing demand assignment MAC protocols.

1.1 Thesis Layout

This thesis presents a study of medium access control in a wireless network. Many issues are required to be taken into consideration when designing a wireless MAC protocol, such as the characteristics of the wireless medium and the service requirements of the traffic being carried. These will be discussed in Chapter 2. In Chapter 3, existing wireless MAC protocols are classified and described in detail. Chapter 4 describes a new protocol called transmission probability based dynamic slot assignment (TRAPDYS). In Chapter 5, the performance of the TRAPDYS protocol is evaluated by using results obtained from stochastic simulation. Conclusions and future work are discussed in Chapter 6.

Chapter 2

Design Issues for Wireless Medium Access Control Protocols

In the first chapter, we described two types of wireless network topology. Both influence the design of a MAC protocol. Two other important issues in the design of wireless MAC protocols are the wireless environment, and the performance and service requirements.

2.1 Wireless Environment Issues

The wireless medium has several unique properties. These make the design of a wireless MAC protocol more challenging than the design of a wired one.

2.1.1 Half-duplex mode

Duplexing is the multiplexing of data transmission and data reception over the same channel. In a wireless environment, it is very difficult to transmit and receive a signal at the same time using the same radio frequency. This is because a large amount of the signal's energy leaks into the receiving path when a node is transmitting. The power of the leakage is usually greater than the incoming signal, making it difficult to detect. A wireless system based on radio frequencies cannot be full-duplexed because the outgoing transmission becomes the interference source for the incoming signal. This is called self-interference. Instead, two half-duplex modes are used: time division duplex (TDD) and frequency division duplex (FDD).

In TDD (Figure 2.1a), one single radio frequency channel is used. The channel is divided in time and is responsible for transmission and reception. Strong temporal organization is required when using TDD.

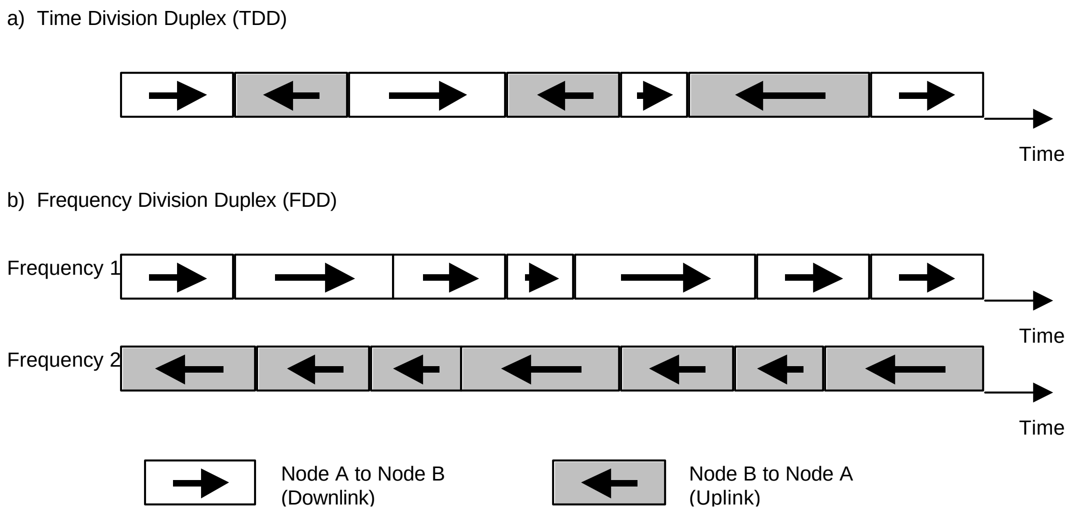

Figure 2.1: Half-duplex mode.

In FDD (Figure 2.1b), two radio frequency channels are used. One is dedicated for transmission and the other for reception. FDD requires a centralized network topology because it is designed for point-to-point communication. The transmission frequency of one node is the receiving frequency of the other. FDD cannot be used in an ad-hoc topology, where the nodes are required to communicate with each other without a central administrator.

A centralized network can have TDD or FDD as its half-duplex mode. If the traffic carried by the network is unbalanced, e.g. Internet browsing where downlink traffic is heavier than uplink traffic, then using TDD is more beneficial to the network. TDD allows an unbalanced traffic flow by readjusting the frame size. Many of the centralized MAC protocols can be converted from TDD to FDD or from FDD to TDD.

2.1.2 Time varying channel

Time varying channel problems are results of radio waves' properties. When a radio wave is transmitted through the air, it propagates based on reflection, diffraction, and scattering. Multiple time-shifted waves are generated when the wave propagates. These waves arrive at the destination at different times, directions, and strengths. A wave is broken into many parts and travels along different paths to the destination. This phenomenon is called multi-path propagation. The physical layer is required to deal with this problem and provide a clear reception. Although the multi-path propagation problem does not have a direct influence on the design of a MAC protocol, it does affect the length of synchronization and propagation delay.

If the receiving signal strength drops to a certain level, fading occurs. Fading can cause some parts of the transmission to be missed, and in the worst case, the entire connection can be lost. In a cellular network, small packets are transmitted between the BS and the MTs to measure the strength of the transmission. If the signal strength of the current BS is weak and the signal strength of the neighbouring BS is strong, a handover is done to the MT.

2.1.3 Burst Channel Errors

Due to the propagation properties of the wireless medium, errors are more likely to take place in the transmission, as there are more sources of interference between the transmitter and the receiver than in the wired environment. In the wired environment, the bit error rates are usually less than 10^{-6} (one error in one million bits), and are mainly due to random noises. In the wireless environment, the bit error rates are usually around 10^{-3} (one error in one thousand bits). These errors usually come in bursts. Long bursts occur when fading occurs. An entire packet can be lost due to burst error.

Many strategies are used to minimize the effect of burst errors. By shortening the length of a packet, the probability of an error occurring in a packet can be decreased. Forward error correcting [Berl87] is a popular strategy against burst errors. If an un-recoverable burst error has occurred in a small packet, the damage is small. Retransmission and acknowledgement are also commonly used to ensure that a packet reaches its destination.

2.1.4 Location dependent effects

Radio signals are usually broadcast into the surroundings except for high frequency bands, which have short, directional wavelengths. Therefore, the location of the transmission and reception becomes important in a wireless environment. There are three location dependent effects: capture, hidden nodes, and exposed nodes, which must be considered when designing a MAC protocol. Centralized network topology can avoid the problem of hidden nodes and exposed nodes.

Capture

The capture effect [Good87] occurs when two or more wireless nodes are transmitting to a target node at the same time. Differences in signal strength can occur due to distance, transmitter power, or interference. Only the node with the strongest signal is received by the target node.

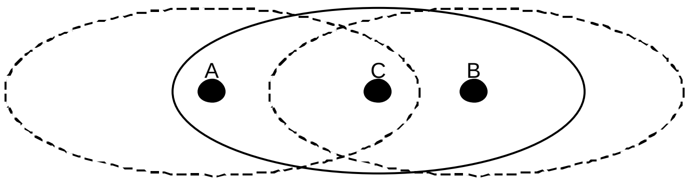

Figure 2.2: An example of the capture effect.

In Figure 2.2, both node A and node B want to transmit signals to node C. Although A and B have the same transmitting power, node B is closer to node C, so the received signal power from node B is greater and is the only transmission picked up by C. The capture effect can be beneficial even though it is unfair. In this example, a collision would have occurred if the capture effect did not take place. The capture effect can be minimized however, by using a sophisticated power controller.

Hidden nodes

The hidden node problem [Bhar98] consists of a hidden sender problem and a hidden receiver problem. The hidden sender problem occurs when a node wants to transmit to a node that is currently receiving a transmission. In the example in Figure 2.3, node A transmits a packet to node B. Node C, a hidden sender, also has something for node B, but it does not hear the ongoing transmission from node A. If node C transmits to node B, a collision occurs.

A control handshake is commonly used to avoid the hidden sender problem. In the same example, node A transmits to node B. Node B broadcasts some kind of signal (or packet) before or while it is receiving a transmission from node A. However, control handshakes can generate a hidden receiver problem. To continue with the above example, node B uses a control handshake to warn the other nodes that it is receiving a transmission. Node C hears the control handshake and defers its transmission. While this is happening, node D has something for node C. Since node D is too far away from node B, it does not hear the control handshake generated by node B. Node D transmits a packet to node C. Node C receives the packet successfully, but cannot acknowledge node D because transmitting an acknowledgement would cause a collision in node B. Node C, here, is a hidden receiver. The hidden receiver problem can be avoided by using out-of-band signalling.

Exposed nodes

The exposed node problem [Bhar98] is similar to the hidden node problem. It is made up of the exposed sender problem and the exposed receiver problem. The exposed sender problem occurs when a node that has something to transmit is exposed by an ongoing transmission. For example in Figure 2.3, node C is transmitting a packet to node D. Node B has a packet for node A but cannot initiate a transmission to node A. The transmission from node B has a chance to collide with the transmission from node C. Node B is an exposed sender. This problem can be avoided by using a receiver-initiated handshake.

Another problem also exists when two nodes are transmitting simultaneously. This is the exposed receiver problem. In Figure 2.3, node C transmits a packet to node D. Node A has a packet for node B. A collision occurs when node A transmits its packet to node B. The transmission from node C collides with the transmission from node A, although the transmission from node C is not intended for node B. Node B is an exposed receiver. The exposed receiver problem is difficult to avoid. Most MAC protocols cannot solve this problem, except for a special type of MAC protocol called the multi-channel MAC protocol (discussed in Chapter 3), which uses several orthogonal frequencies for different nodes to transmit their packet.

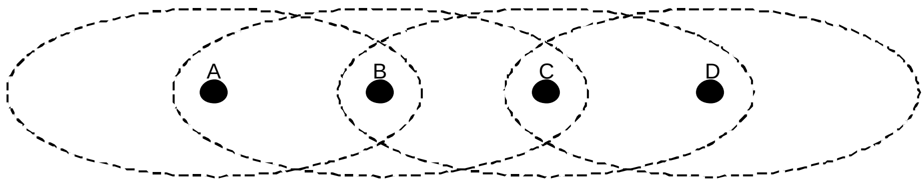

Figure 2.3: An example of a hidden and exposed node problem.

2.1.5 Wireless technologies

The wireless technologies used in the physical layer can also influence the design of a MAC protocol. Code division multiple access (CDMA) and orthogonal frequency division multiple access (OFDM) allow many nodes to transmit at the same time using the same radio frequency. In CDMA and its variations [Pras98], each transmission is coded and spread over a wide radio band. The receiver uses a code to de-spread the transmission and recover the data. There are three basic types of CDMA: Direct Sequence CDMA (DS-CDMA), Frequency Hopping CDMA (FH-CDMA), and Time Hopping CDMA (TH-CDMA). In DS-CDMA, each transmission is coded with a code that has a higher density than the data. In FH-CDMA, each code is a small piece of the original transmission and is transmitted at a different frequency. The code hops from one frequency to another, using only a small part of the frequency band in each hop. The TH-CDMA is coded along a time axis. The code hops along a time line and does not transmit continuously. By using the same code, the destination can extract the original data.

In satellite technology, the long propagation delay between the nodes must be accounted for in the design of a wireless MAC protocol. Smart antenna technology [Lehn99] allows spatial multiplexing. The design of a MAC protocol must take into account the additional dimension.

2.2 MAC Protocol Performance Issues

There are several issues important to a MAC protocol that are also metrics for measuring its performance. Most of them apply to both wired and wireless MAC protocols, although some are wireless specific. The following is a brief discussion of these issues:

Delay

Delay is usually measured as the average time a packet spends in the queue. It is the time from the generation of a packet until it is transmitted successfully. The delay requirement is usually dependent on the type of traffic being transmitted. A long delay is not always bad. If the payload of each transmission is large, then the overall transmission rate is not affected. If the traffic being carried is delay sensitive (e.g. real time voice), then a long delay is not suitable.

Throughput

Throughput is the fraction of the total channel capacity that is used for data transmission. If the throughput is high, then the bandwidth wastage is small. The goal of a good MAC protocol is to maximize throughput, while minimizing access delay.

Fairness

A fair MAC protocol ensures that each node has the same opportunity to access the channel. Unfairness can produce dominators in the system and cause the system to be unpredictable and unbalanced. An example is the capture effect. If a node close to the target node has a large amount of data to transmit, then the node further away from the target must wait for a long time before transmitting its packets to the target node.

Stability

A network is required to be stable at all times. This includes the occasional heavy load on the channel that is greater than the maximum transmission capacity. An unstable MAC protocol can fall apart during a heavy load. The average delay can rise dramatically and cause the channel to be jammed for a very long period. A stable system should handle heavy loads without a long delay.

Power Consumption

Most mobile devices are small, light, and easy to carry. However, their battery power is usually limited, so it is important to design a MAC protocol that consumes little power. A MAC protocol can save power by grouping the broadcasts. This way a node only needs to power up to listen to the broadcast for short periods.

Multimedia support

With the increasing popularity of multimedia such as image, voice and video, wireless networks need to support multimedia traffic. Video traffic has a constant bit rate, that is time sensitive, and voice traffic has a variable bit rate, that is also time sensitive. It switches from on to off and off to on. To support these two types of traffic and provide a good quality of service (QoS), a MAC protocol must satisfy their delay requirements by allowing some kind of prioritisation scheme.

2.3 Frame, Phases, Slots, and Channels

Finally, let us define the basic entities used when discussing MAC protocols: frames, phases, slots, and channels. Figure 2.4 shows the layout of an uplink stream of a demand assignment MAC protocol. The uplink stream is divided into frames of equal length. A frame is a time interval structure consisting of many elements such as phases and slots. There are usually two types of frames: one uplink and one downlink that follow one another successively. A frame has many phases. A phase is an action of the protocol over a time period in a frame. The example in Figure 2.4 shows uplink frames with two phases. The reservation phase is a time interval that is used by MTs to send their requests to the BS. The data transmission phase is the part of the protocol in which the uplink data transmissions occur. Some phases consist of control information such as synchronization bits, but most consist of slots, time intervals of predefined lengths. In such an interval, a data block called a packet can be transmitted from point A to point B. Different types of packets have different lengths and require different slot sizes. There are usually two types of slots in an uplink frame: request slots and data slots. A request slot has a shorter time interval. It is just big enough to transmit a request (request packet). A data slot is much longer and is used to carry the actual data (data packet). The reservation phase consists of request slots and the data transmission phase consists of data slots.

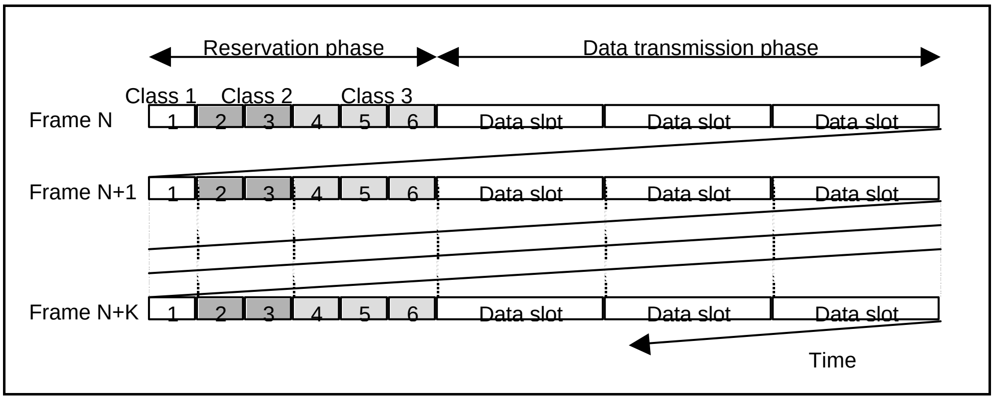

Figure 2.4: Different phases, slots, and channels in a demand assignment MAC frame

A channel can be a frequency band, a collection of frames, different phases in a frame, slots, or any other resource in time or frequency domain used to send and receive information. For example in Figure 2.4, the reservation phase as a collection of frames in time can be considered as a channel. It is often called a random access channel (RACH), see discussion of a global system for mobile telecommunication (GSM) [Rahn93]. This channel sits in the beginning of each frame. During the data transmission phase, a channel similar to that of the reservation phase is used. Both channels contain many slots. In Figure 2.4, the request slots are grouped together to create classes. Each group is assigned to a different class of traffic. Here the grouping occurs continuously in the following frame. We can consider each one of these groups as a channel.

2.4 Summary

Many issues must be considered when designing a good wireless MAC protocol. These can be divided into two types. The first are those generated by the wireless medium. They are heavily associated with the transmission arrangement and the integrity of transmission in the wireless medium. The second are concerned with efficiency and performance. Both issues are commonly found in all MAC protocols, either wireless or wired. In the next chapter, we discuss how these issues affect the design of the wireless MAC protocol and present a detailed survey.

Chapter 3

Wireless Medium Access Control Protocols

In 1970, a pioneer MAC protocol for radio communication called Aloha [Abra70] was proposed. It was one of the first protocols to control the access to radio medium. Since then, hundreds of MAC protocols for wired and wireless systems have been published. Today wireless MAC protocols are much more complex and able to support multimedia traffic and provide a good QoS. However, many still use Aloha in some of their phases. In Section 3.1, we will provide an overview of more representative MAC solutions often used in various phases of newly published protocols. In Section 3.2, we discuss the classification of wireless MAC protocols and provide examples.

3.1 Fundamental Medium Access Control

In this section, a brief overview of the fundamental techniques and MAC protocols important to MAC design is discussed. These protocols do not include features such as prioritisation of mobile terminals and QoS requirements, but their ideas are relevant.

Aloha

The Aloha access protocol [Abra70] was proposed in 1970. It is very simple and fundamental. Each node in the system is completely independent. When a node has generated a new packet, it is transmitted immediately. The node does not need to observe the channel before transmission. After transmission, the node waits for an acknowledgement from the receiver. If no acknowledgement is received within a predefined period, the transmitted packet is assumed to have collided or been lost. The node then enters a collision resolution state, in which it waits for a random time before retransmitting the packet. The Aloha access protocol provides completely free access to the radio channel, and can be used on any network topology.

Slotted-Aloha

The slotted-Aloha protocol [Robe75] is an improvement on the Aloha protocol. It is widely used as the random access method for the more complex protocols. In the slotted-Aloha protocol, the time axis is divided into time slots. Each time slot is equal to the transmission time of a packet. When a new packet is generated by a node, it is transmitted in the next time slot. Similar to Aloha, the node waits for an acknowledgement from the receiver. If the acknowledgement is not received after a predefined period, then the node assumes that a collision has occurred. The node backs off (remains silent) for a random number of slots before retransmitting the packet. The slotted-Aloha protocol is more efficient than the Aloha protocol, as when a collision occurs, only one time slot is wasted. The entire transmission (which could be several time slots in size) is lost when a collision occurs in the Aloha protocol. However, the slotted-Aloha protocol is not as unrestricted as it requires time slots to be synchronized between the nodes.

p-persistent

The p-persistent access protocol is similar to the p-persistent CSMA [Klei75] protocol and the slotted-Aloha protocol. The time is divided into time slots. When a node has a packet to send, it draws a random number. If the random number is greater than p, the packet is transmitted in the next available slot. If the number is less than p, the node repeats the random number process again in the next slot. The process will continue until the packet is sent successfully.

Collision Resolution Algorithms -Binary Exponential Backoff

The binary exponential backoff (BEB) is one of the most commonly used collision resolution algorithms. The protocol is easy to implement, does not require many hardware resources, and can work on top of the slotted-Aloha protocol. BEB increases the backoff time of collided nodes to relieve the traffic. When a transmission from a node collides, the node backs off. If a collision occurs again for the same packet, the backoff time is doubled. The backoff time of the node doubles each time the same packet collides.

Collision Resolution Algorithms -Tree Algorithm

Tree algorithms can usually provide highly efficient collision resolution in a time slotted channel. Much research has been conducted in this particular area. The recently standardized IEEE 802.14 standard for hybrid fibre coaxial cable TV networks (HFC-CATV) uses a highly optimised ternary tree algorithm [Sala98] for collision resolution. The multi-accessing tree protocol [Cape79] was one of the first tree algorithms proposed in the 1970s. The algorithm works on top of the slotted-Aloha protocol. When a collision occurs, the tree algorithm is used. The nodes that are not involved in the collision are required to wait until the collision is resolved before transmitting their packets. The collided nodes pick 0 and 1 randomly with a probability of ½ before a time slot. If a node has chosen a 1, it increments a special transmission counter by 1. When the node observes either a successful transmission or an empty slot it decrements the counter by 1. The node transmits its packet when the counter reaches zero.

3.2 Wireless Medium Access Control Protocol

The problems faced by a wireless network are very different from those of a wired network. Many papers have been produced in these related areas over the last thirty years. They can be classified into three groups. The first are papers that have proposed new MAC protocol. The second report the performance evaluations and comparisons of pre-existing MAC protocols. The third combine different network environments and technologies with a pre-exiting MAC protocol. In this section, we will look more closely at the first group. Their methods will be discussed and each protocol classified based on their approach.

3.2.1 Classification

Capetenakis in [Chan00] have logically classified wireless MAC protocols based on the network topology used and their access method. We have expanded their classification tree and created a new branch, see Figure 3.1. The protocols are first classified based on the network topology they use: ad-hoc, centralized, and ad-hoc centralized. The ad-hoc topology and the centralized topology were discussed in Chapter 1. The ad-hoc centralized topology is a combination of the two. It is a mobile cell that can be formed anywhere. An ad-hoc centralized network looks like an ad-hoc network on the surface, but it is centralized underneath.

MAC protocols for ad-hoc networks can be classified further into busy-tone based and collision avoidance based. The protocols in both classes avoid collisions by sending out some kind of signal or packets.

MAC protocols for centralized networks can be divided into three classes: random access, guaranteed access, and hybrid access. The protocols in the random access class try to obtain high efficiency while maintaining simplicity. The protocols in the guaranteed access class eliminate collisions by using polling. The hybrid access class can be further divided into two classes: random reservation and demand assignment. The random reservation protocols require and reserve the bandwidth in a free fashion. The demand assignment protocols transmit a request and wait for the BS to assign a bandwidth to them.

The following sections will discuss the MAC protocols from each class in greater detail.

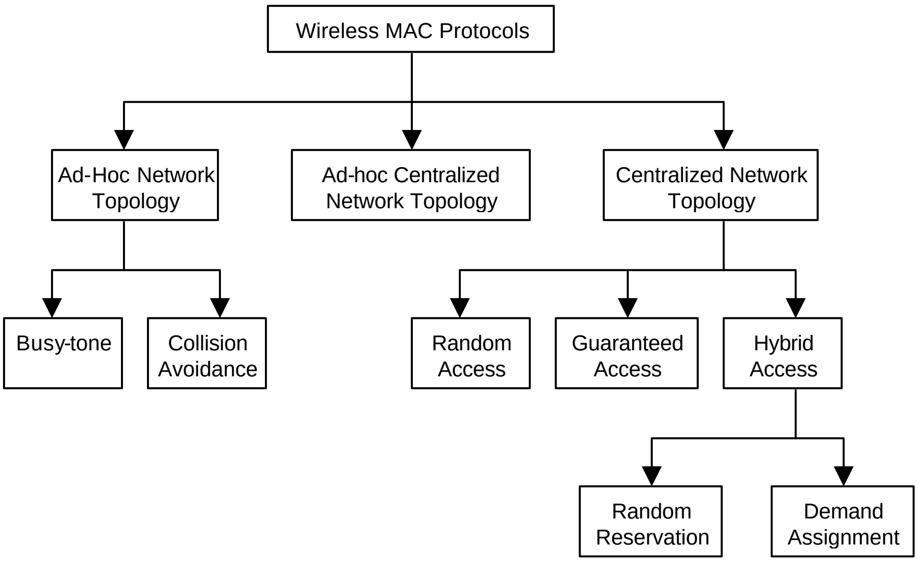

Figure 3.1: Wireless MAC protocol classification tree.

3.2.2 Wireless Medium Access Control Protocols for Ad-hoc Networks

An ad-hoc network has a decentralized structure. The network is made up of a collection of nodes. Their network function is similar to each others. The nodes have to communicate with each other without any pre-existing infrastructure. MAC protocols designed for ad-hoc networks must take into account their shapeless structure. Two approaches have been used: the first is called busy-tone, and the second is known as collision avoidance. These are discussed below.

3.2.2.1 Busy-tone protocols

Busy-tone protocols use an out-of-band busy-tone signalling. A very narrow frequency band (or frequency channel) is used to carry the busy-tone signal. It does not interfere with the data channel. The busy tone is a signal to warn the surrounding nodes not to transmit. The nodes in the network are required to listen to the busy-tone channel before they transmit any packet. If there is an assertion on the busy-tone channel, they defer their transmissions to a later time according to the scheme used by their protocol. The node that is engaging in a transmission usually asserts a busy tone in the busy-tone channel. The following are examples of busy-tone based MAC protocols.

Busy Tone Multiple Access (BTMA)

BTMA [Toba75] is one of the earliest busy-tone protocols. The protocol has two channels, one for busy-tone signalling and the other for data transmission. When a node has data to transmit to a neighbouring node, it transmits its data packet in the data channel according to the slotted-Aloha protocol. Any neighbour, including the receiver node that hears this ongoing transmission, transmits a busy tone in the busy-tone channel. Once a node hears this, it backs off until the busy tone is over. The busy tone creates a double radius inhibition zone, and all nodes within this zone are inhibited from transmission. This eliminates the hidden nodes that surround the host node and the target node, but increases the number of exposed nodes.

An improved version of BTMA called Receiver Initiated-BTMA (RI-BTMA) [Wu88] was designed to combat the large number of exposed nodes produced by the BTMA protocol. In RI-BTMA, the host node sends a short message to the intended target. Any node that hears this ongoing transmission decodes the short message and then identifies the intended receiver. If the node is the intended receiver, it broadcasts a busy tone in the busy-tone channel. The busy tone acts as an acknowledgement to the host node. The host can then begin its data transmission. The other nodes back off after hearing the busy tone. RI-BTMA decreases the number of exposed nodes.

Wireless Collision Detect (WCD)

The WCD protocol [Gumm00] is designed for a short radius network (< 50m). The frequency channel is split into a data-channel and a feedback channel. The feedback channel consists of two logical channels called the carrier detect (CD) channel and the feedback-tone (FT) channel. The assertion of the two logical channels does not occur simultaneously.

A node is in either the data reception mode or the data transmission mode. In the data reception mode, the node listens to the data-channel. When it detects a transmission, it asserts on the CD channel. After the header is received, the node determines the destination of the packet. If the destination address matches its own address, the node stops asserting on the CD channel and asserts on the FT channel. If the address is not matched, the node simply stops asserting on the CD channel.

When the node is in the data transmission mode, it samples the CD channel and FT channel before transmission. If the node detects an assertion on the FT channel, it will back off for a period. If it detects an assertion on the CD channel, it will sample the channel again after a Receiver Detection Interval (RDI). An RDI is the time required to determine the destination of the current transmission and assert on the feedback channel. When there is no assertion detected on either channel, the node will sense the CD channel for an Idle Detection Interval (IDI). An IDI is the round trip time plus the time to detect a carrier plus the time to assert the feedback signal. If the CD channel is not asserted during that period, the node makes a transmission attempt. After an RDI, the node will sample the FT channel for feedback. If the FT signal is not asserted, the node will assume that a collision has occurred. It will abort its transmission and back off for a random period before attempting to transmit again.

Consider the example of the four nodes shown in Figure 2.3. When node A is transmitting to node B, node B asserts an FT signal. Although node C cannot hear the ongoing transmission between node A and B, it can detect the assertion, and will not transmit any data. If node D is transmitting data to node C at this time, node C is still able to acknowledge the transmission from node D by asserting on the FT channel. This eliminates the hidden sender and hidden receiver problem.

The protocol also eliminates the exposed sender problem. Consider the following example. Node B transmits data to node A and node A asserts on the FT channel. Since node C is too far away to hear the FT signal from node A, it can send data to node D without any interference from node B.

3.2.2.2 Collision avoidance protocols

The collision avoidance approach avoids collisions by using control handshakes. These handshakes are short packets carrying messages to inform the surrounding nodes. The handshakes are similar to the busy tone, but carry more information. Three handshakes are commonly used by the collision avoidance protocols: request to send (RTS), clear to send (CTS), and acknowledgement (ACK). RTS is usually sent by a host node to a target node. The purpose is to inform the target node that the host node has something to transmit, and also to ensure the target node is free and avoid collisions. CTS is used by the target to reply to the host after receiving an RTS. ACK is used simply to inform the host that its data transmission has been successful. The handshakes are also used to warn the surrounding node that a transmission is ongoing.

The collision avoidance protocols usually operate in a single channel mode (single frequency band), with handshakes being exchanged in the channel. Multi-channel protocols also exist. In these protocols, multiple frequency bands are used for handshakes and data transmission, and greater organization is required. The following are some examples.

Multiple Access with Collision Avoidance (MACA)

MACA [Karn90] uses a two-way handshake mechanism to avoid collisions. When the host node wants to transmit data to the target node, it sends an RTS packet to the target node. Any neighbouring node of the host defers its transmission when it hears the RTS. If the target receives the RTS successfully, it responds by broadcasting a CTS packet. The CTS warns the neighbours of the target node not to transmit. When the host receives the CTS, it assumes that the channel is clear and sends its data to the target node.

Floor Acquisition Multiple Access (FAMA)

FAMA [Full95] is a protocol based on collision avoidance and handshakes. A node must acquire the surrounding channel (“floor”) before transmitting its data. To acquire the floor, a node transmits an RTS to its neighbours. When the target node receives the RTS, it responds with a CTS message if it is free. The host node then begins sending its data packets. The CTS also serves to warn off other nodes from transmitting to the target node.

Floor Acquisition Multiple Access with Non-persistent Carrier Sensing (FAMA-NCS) [Garc99] improves on FAMA. Its goal is to provide better collision avoidance by changing the length of the CTS. FAMA-NCS uses a long CTS message.

The length of a CTS is the time required for transmitting an RTS message, the maximum roundtrip time, the turn around time, and the processing time. When a target node has begun to transmit a CTS, its neighbours that are transmitting an RTS will receive at least a portion of the CTS and back off. This allows the host node to transmit its data packets without collisions from the neighbours of the target node.

Distributed Foundation Wireless MAC (DFWMAC)

DFWMAC [Crow97] is the basic access protocol for distributed systems described by the IEEE 802.11 standard. Four handshakes (RTS-CTS-DATA-ACK) are used between two nodes in this protocol. The host node that wishes to transmit data to the target node first senses the channel idle for a period of time before attempting an RTS transmission. This period is called the DCF Inter-Frame Space (DIFS) (DCF stands for distributed coordination function). If the channel is busy during this period, the host node backs off for a specified interval. The host node transmits its RTS after DIFS is finished. When the target node receives the request from the host node, it senses the idle channel for a Short Inter-Frame Space (SIFS) before sending a CTS message. Once the host receives the CTS, it waits for SIFS, then begins to transmit its data. If the data is received by the target successfully, the target waits for SIFS then sends an ACK. SIFS is shorter than DIFS. This provides a priority scheme in favour of transmission attached to SIFS. The inter-frame space is used as a form of collision avoidance.

The neighbouring nodes listen to the channel traffic and predict the length of the transmissions based on virtual carrier sensing. A Network Allocation Vector (NAV) is used in virtual carrier sensing. It is a period indicating the time required to wait before transmission. When a neighbouring node hears an RTS from the host node, it expects the host to target transmission to complete in an NAV (RTS) interval. If it hears a CTS, it expects the transmission to complete at the end of an NAV (CTS).

Broadcast Support Multiple Access (BSMA)

The BSMA protocol [Tang00-1] is an extension of the IEEE 802.11 protocol. Its goal is to provide an efficient broadcasting ability. It incorporates the collision avoidance scheme and the four handshakes control of IEEE 802.11. It relies on negative acknowledgement (NACK) to deliver broadcast packets.

The host node that has a packet to broadcast first goes through the collision avoidance phase identical to that in IEEE 802.11. The host then sends out an RTS to its neighbours and sets the WAIT_FOR_CTS timer. If the timer expires before the host receives a CTS, it repeats the step. After successful reception of the RTS, the neighbouring nodes that are not in a prohibited state transmit a CTS message and set the WAIT_FOR_DATA timer. Any node that receives this CTS message and is not part of the transmission changes its state to a prohibited state until the end of this transmission predicted from the NAV. Upon receiving the CTS message, the host sends its data and sets the WAIT_FOR_NACK timer. When a neighbouring node does not receive the data successfully from the host before the WAIT_FOR_CTS timer expires, it transmits a NACK to the host node. If the host does not receive any NACK before the WAIT_FOR_NACK timer expires, it assumes that the transmission has been successful.

Adaptive Broadcast (ABROAD)

In the ABROAD protocol [Chla00], the frequency is divided into frames. Each frame consists of N sub-frames, where N is the number of nodes in the network. The sub-frames are assigned to the nodes in a one-to-one fashion. An MT has priority to use the sub-frame that is assigned to it in a frame. A sub-frame is made up of five periods including four signal periods and one data period.

When a node has a broadcast packet to send in its assigned sub-frame, it transmits a request-to-broadcast (RTB) packet in the first period of the sub-frame. Each neighbouring node responds with a clear-to-broadcast (CTB) packet in the second period after it received the RTB packet. This informs all nodes in a two-hop radius not to transmit in that sub-frame. The central node waits for the other two signal periods to pass through. If an idle is observed in the two signalling periods, the node broadcasts its packet.

If the node that is assigned to the ongoing sub-frame does not have anything to transmit, an idle is observed in the first two periods. When this occurs, a node with data to transmit can use the sub-frame. It does this by transmitting an RTB packet in the third period of the sub-frame. If its neighbours detect a collision, they send a negative-CTB (NCTB) packet in the fourth period. The node transmits its data packet when no NCTB packets are detected. This protocol eliminates the hidden sender problem but not the hidden receiver problem. The exposed sender problem does not exist in a broadcasting protocol such as this.

This protocol focuses on broadcasting only. The protocol claims to support unicast service but the author did not provide a way to achieve this claim. No feedback mechanism was included in the paper. In an ad-hoc network, the number of nodes is likely to change from time to time. The number of the sub-frames in a frame changes when the number of nodes in the network changes. The protocol did not provide a method of coping with this variation.

Dynamic Channel Assignment (DCA)

DCA [Wu00] has been designed for a multi-channel network. The single channel protocol engages only one channel for information transmission between all nodes. The channel is usually a frequency band. In a multi-channel network, multiple channels are employed for information transmission. Depending on the technology used, the channels can be frequency bands or CDMA codes. Commonly a channel is assigned to several nodes in the network. To communicate with a node, the source has to transmit its information using the channel that is assigned to the destination node. The other nodes cannot pick up the transmission unless they too have the same assigned channel.

In DCA, the overall bandwidth is divided into one control channel and N data channels. The control channel is used to resolve contention in the data channels and to obtain access rights to the data channels. Each node in the network maintains two lists: the channel usage list keeps information about the neighbours and their channel usage; the free channel list is computed by the node from the channel usage list.

When a host node wants to communicate with a target node in the neighbourhood, it sends an RTS with its free channel list to the target node through the signalling channel. The target node matches the incoming list with its own channel usage list to identify a data channel to be used. It then replies to the host node with a CTS message. The CTS warns the surrounding nodes of the target node not to use the channel. After receiving the CTS from the target node, the host node transmits a reservation packet to inhibit other neighbours from using the same channel. The host then begins to transmit its data packet to the target node.

A set of complex rules and calculations are used after a node receives the free channel list that comes with the RTS. These rules help to schedule the transmission and avoid collisions.

Collision-Avoidance Transmission Scheduling (CATS)

CATS [Tang00-2] is a multi-channel protocol for an ad-hoc network. The protocol is an extension of CATA [Tang99]. CATS is designed to support multicasting and broadcasting. It uses a negative feedback mechanism. The bandwidth is divided into one signalling channel (SCH), one broadcast data channel (BCH), and N data channels (DCH). The channels are divided into time slots. Each slot is further divided into five mini-slots (MS1 to MS5) and a data slot. There are seven types of signalling messages, called beacons, used in the SCH. The size of a beacon is the same as a mini-slot. Each node in the network has to listen to the SCH when it is not engaged in reserving a channel, or sending or receiving data over a channel.

Every node that is going to send or receive data in the current slot transmits a link reservation beacon (LRB) in the MS1 over the SCH. This is to notify the neighbours that it is busy. In the case of unicast and broadcast, every receiver node has to transmit an LRB in MS2. This is to warn its neighbours not to attempt to establish a multicast or unicast link over it. MS3 to MS6 are for data transmission in the BCH and DCH.

In a reservation for unicast, the host node listens to MS1 in the SCH to make sure the slot is idle. The host then listens to the MS2 in destination DCH. If the destination node is silent, the host transmits a request unicast beacon (RUB) over the SCH during MS3. After the destination successfully receives the RUB, it listens to the DCH during MS4. If a silence is observed in MS4, the destination sends a concur-with-unicast beacon (CUB) in MS5. Once the host node receives the CUB, it begins to transmit its data in the rest of the time slot and the same slot of the upcoming DCH frames.

A similar procedure is used in a reservation for multicast. A request multicast beacon (RMB) is used instead of a RUB during MS3. The host remains quiet during MS5. If the host finds that the SCH stays clear during MS4 and MS5, it assumes that the reservation is successful and transmits its data in MS6 over the DCH.

For a broadcast reservation, the host sends a request broadcast beacon (RBB) during MS3 after observing a silent MS1. If a node receives RBB or observes a silent period during MS3, it remains silent during MS4. Otherwise, the node transmits a stop broadcast beacon (SBB). The host assumes that the reservation is successful if it does not detect SBB during MS4.

3.2.3 Wireless Medium Access Control Protocols for Centralized Networks

The centralized MAC protocols are designed for a centralized wireless network. The network consists of a base station and many mobile terminals. The BS sits in the middle of the cell and administers all the traffic. This type of topology allows a highly optimised medium access control.

3.2.3.1 Centralized Random Access Protocols

The random access protocols provide a high level of flexibility. The MTs can access the channel freely and randomly. The major source of inefficiency in the random access protocols is packet collision. Allowing the MTs to transmit freely and randomly means no order or guarantee is given to the system. A collision occurs when two or more MTs pick the same slot to transmit their data.

Idle Sense Multiple Access (ISMA)

In ISMA [Wu93] (Figure 3.2a), the BS broadcasts an idle signal (IS) when the channel is idle. There is a propagation delay between each IS. This delay allows the BS to obtain responses from the MTs that are intending to transmit in the following period. The BS listens to the traffic and does not transmit another IS if there is an ongoing transmission. An MT can transmit its data packet to the BS when it has listened to the channel and heard that the channel is idle. The size of the data packet is limited, so an MT can only transmit a certain amount of data each time. If the data packet from an MT has been transmitted and received by the BS without any error or collision, the BS then transmit an idle signal with an acknowledgement (ISA). This acknowledges the successful packet transmission and allows other MTs to know that the channel is free again. If, unfortunately, a collision has occurred, the BS does not do anything about the collision, but transmits an IS after it. The transmitting MTs detect the collision after they receive an IS instead of an ISA. The MTs then back off for a random period before trying again.

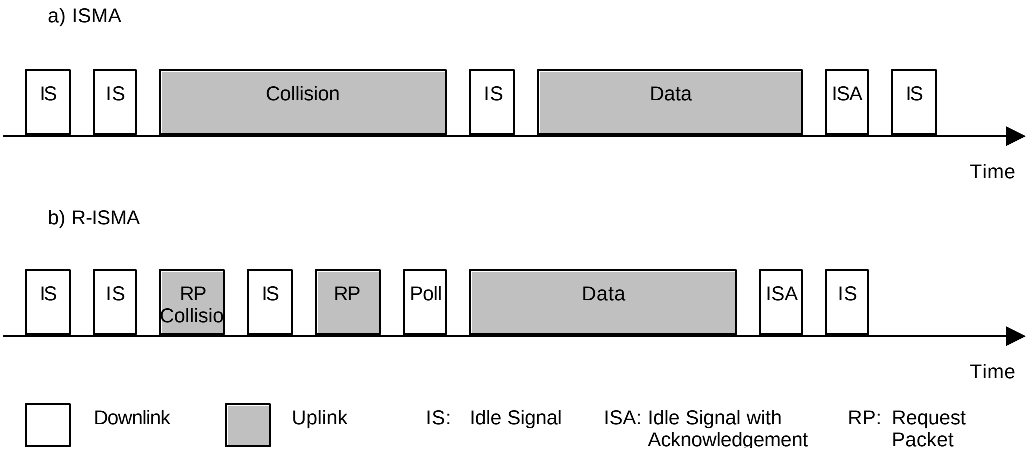

Figure 3.2: Idle sense multiple access protocol.

When a collision occurs in ISMA, the entire data packet is lost. To minimize this damage as much as possible, an improved method called Reservation ISMA (R-ISMA) [Wu96] (Figure 3.2b) was proposed. This protocol uses a reservation packet and polling to minimize the damage caused by a collision. After sensing the idle channel, the MT transmits a short reservation packet instead of a data packet. If a collision occurs when transmitting the reservation packet, the MT retransmits the packet after a random period. If the transmission is successful, the MT waits for a poll from the BS. After receiving the reservation packet, the BS polls the MT by sending out a polling signal. Upon receiving the polling signal, the MT immediately transmits its data packet. During this time, there are no IS transmitted by the BS, therefore, no MTs will attempt a transmission. If the data packet is received by the BS successfully, the BS broadcasts an ISA; if not it polls the MT to retransmit the data packet.

Randomly Addressed Polling (RAP)

RAP [Chen93] (Figure 3.3) is a protocol that combines contention random access and polling. CDMA is used in the contention phase. When an MT has something to transmit, it randomly picks a CDMA code and uses that as a request in the contention phase. The contention period is situated at the beginning of a frame. Many MTs can transmit their requests at the same time. If two or more MTs use the same code, the BS treats the transmissions as a single transmission. No collision occurs at this stage. The BS then polls each code received in the contention period one by one. When an MT hears the poll of the code it has used previously, it transmits its data packet. If the data packet is received by the BS successfully, the BS sends an acknowledgement immediately. If two or more MTs have used the same code in the contention phase, then a collision occurs when the BS polls the code for the data. The BS sends a negative acknowledgement when it polls the next code. The MTs that have received this negative acknowledgement retransmit their requests in the next contention period.

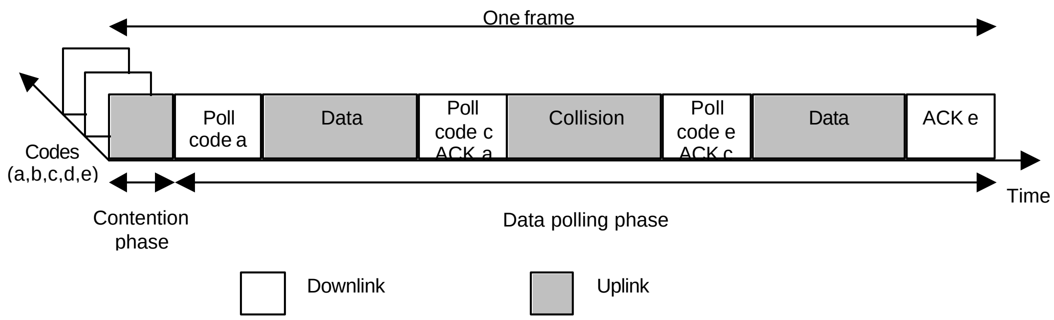

Figure 3.3: Randomly addressed polling protocol.

Resource Auction Multiple Access (RAMA)

RAMA [Amit93] (Figure 3.4) uses a deterministic algorithm for random access. Two phases exist in RAMA: the contention phase and the data phase. The MTs contest in the contention phase to gain access in the data phase that follows. Only one winner is produced in a contention phase. Each MT in the system holds a unique binary ID string. When an MT has something to transmit to the BS, it sends its ID string symbol-by-symbol in the contention phase. The BS broadcasts what it hears after each symbol. If the broadcast from the BS matches the ID string of the MT, the

MT wins the contest. Otherwise, the MT backs off immediately. The MT retransmits its ID in the next contention phase. The winner transmits its data in the data phase.

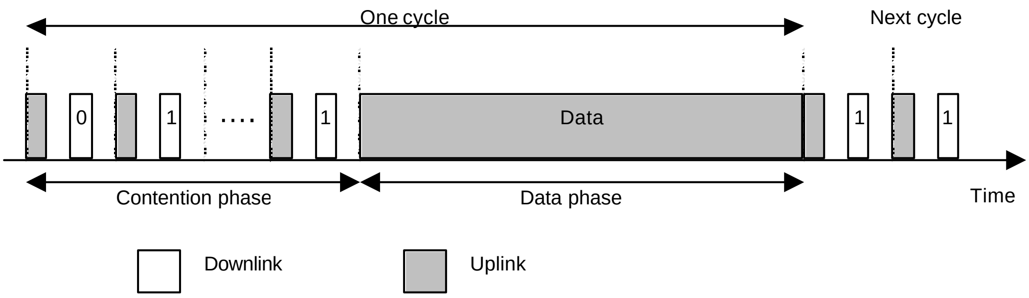

Figure 3.4: Resource action multiple access protocol.

The contest in the contention phase is fixed. When more than one MT transmit the same symbol, no collision results. The BS can broadcast what it hears. If they transmit different symbols, a collision results. In this situation, the BS gives one symbol a higher priority than the other. For example, one MT transmits a “0” symbol and the other MT transmits a “1” symbol. The BS hears a collision and broadcasts “1” in response. The MT that has transmitted a “1” symbol has higher priority. This is unfair to the other MT. Due to this unfair prioritisation scheme, the contest can always produce one winner in the end even if collisions exist. The bandwidth is not wasted on collisions.

3.2.3.2 Guaranteed Access Protocols

Guaranteed access protocols are based on polling. Polling is a control handshake similar to the handshake used in the collision avoidance ad-hoc network. It is initiated by the BS using a small packet that carries a message to a specific MT. Once the MT receives this packet, it responds to the BS according to the protocol it uses. The BS polls each MT in the network for data, one after the other in a round robin fashion. No collision exists in a guaranteed access protocol. These protocols can achieve high utilization when many MTs are accessing the channel. The bandwidth can be wasted when the polled MT has nothing to transmit. If the protocol is not carefully designed, the bandwidth can also be wasted through a large amount of propagation delay when polling.

Zhang's protocol

Zhang's protocol [Zhan91] (Figure 3.5) has two polling phases. In the first phase, the BS polls for requests from each MT in a round robin fashion. If the polled MT has something to transmit, it transmits a request packet. If it does not, it transmits a "Keep Alive" message to let the BS know that it is still there. After each MT has been polled for requests, the second polling phase begins. The BS polls MTs for data in this phase. The MTs that have transmitted a request packet in phase one are polled one by one by the BS. An MT transmits its data packet when it is polled. The downlink data transmissions occur after the two polling phases before the next polling cycle begins. The protocol is very simple, but it guarantees that all MTs are polled. All transmissions are free of collision.

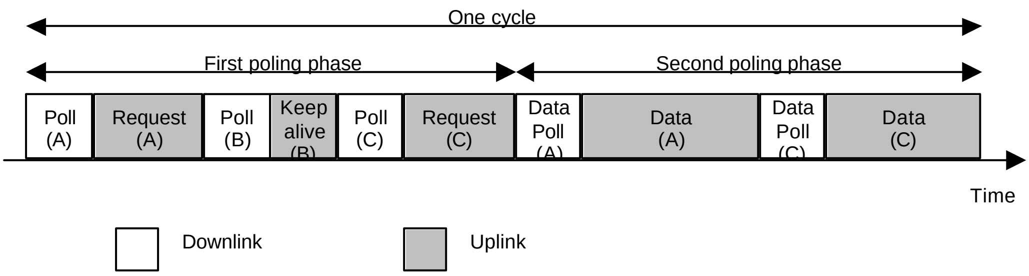

Figure 3.5: Zhang's protocol.

Disposable Token MAC Protocol (DTMP)

In DTMP [Hain93] (Figure 3.6), the data transmissions for the uplink and the downlink are followed by a single poll. The BS polls all MTs one by one. Two types of poll exist in the protocol: the normal poll and the data poll. The normal poll occurs when the BS does not have any data for the polled MT. If an MT is polled, it stays silent. The BS waits for the data for a short period. If no data is transmitted from the polled MT in that period, the BS moves on and polls the next MT. If the polled MT has data to transmit, it transmits its data immediately when polled. The BS replies with an acknowledgement after the data packet has been successfully received.

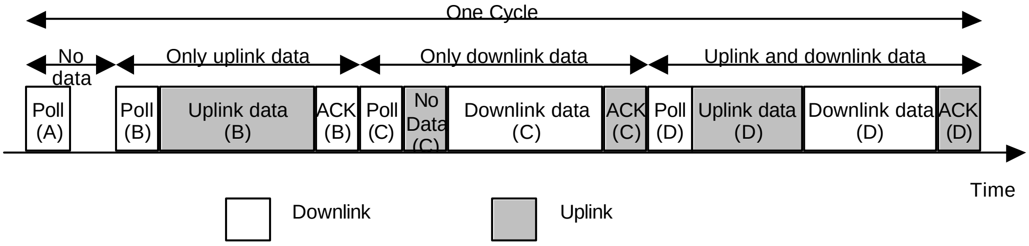

Figure 3.6: Disposable token MAC protocol.

A data poll is used when the BS has something for the polled MT. If the polled MT does not have any data for the BS, it transmits a "no data" message to the BS when it is polled. If the MT has data to send, it sends its data when it is polled. After receiving the "no data" message or the uplink data, the BS transmits its data to the MT. If the downlink data transmission has been successful, the MT sends an acknowledgement. The BS then begins to poll the next MT.

Acampora's protocol

Figure 3.7 is the frame layout of Acampora's protocol [Acam97]. A frame is divided into three phases: polling phase, request phase, and data phase. The polling phase and data phase are based on the polling method. The request phase is based on observations in the polling phase.

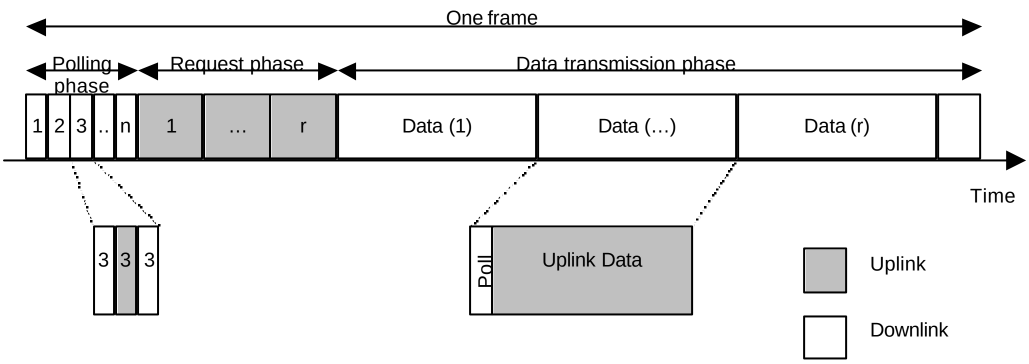

Figure 3.7: Acampora's protocol.

In the polling phase, each MT is polled with a mini-poll. The polled MT replies to the BS with a short message if it has something to transmit. The BS then replies with a short message for confirmation. The main purpose of the polling phase is to inform the BS how many MTs have data to transmit. After polling all the MTs in the network, the BS knows exactly how many MTs want to transmit. The BS then sets the number of request slots in the request phase based on that number. The MTs with packet to send listen to each response in the polling phase. They use this information to calculate how many MTs have requested before them and hence how many slots they have to wait before transmitting their request in the request phase. Using the dedicated slots in the request phase, the MTs transmit their request. The size of the request packet is fixed.

The data phase begins when the last packet in the request phase has gone through. In this phase, the BS polls the MTs that had transmitted in the request phase. The polled MT transmits its data packet when it is polled. After all the MTs that had requested earlier transmit their data, the BS transmits downlink data to the MTs in the network.

The protocol uses a mini-poll to minimize bandwidth wastage when the polled MT has nothing to transmit. It can achieve high efficiency when the number of active MTs is high.

3.2.3.3 Random Reservation Access

The random reservation protocols attempt to combine the flexibility of random access with the guarantees of polling access. At the same time, the protocols must remain simple. Only a small number of protocols can be categorized into this class. More complex random reservation protocols usually contain a demand assignment access feature and hence they are categorized as demand assignment protocols. The random reservation protocol contains two stages: random access and reservation. The MTs access the channel in a free and random fashion. If the accesses has been successful, the MTs then reserve the same slot in the following frames.

Packet Reservation Multiple Access (PRMA)

PRMA [Good89] (figure 3.8) focuses on bandwidth reservation for voice traffic. Frequency division duplex is used. The uplink stream is divided into frames, each containing a number of data slots. The data slots can be accessed by the MTs using the p-persistence slotted-Aloha access scheme. If a slot is not reserved, an MT has a probability of p to transmit in the slot. Two types of traffic are defined in PRMA: voice and data. When an MT has voice packets in its buffer, it transmits the first packet based on the slotted-Aloha protocol in the unreserved slots by listening to the downlink stream. If there is a collision, the MT backs off and tries again later. If the transmission is successful, the same slots of the following frames are reserved for that voice traffic only. The assignment of the slot takes place through the downlink broadcast. The MT can transmit its voice packets without any possibility of collision. The slots are reserved until the end of the talk spurt (where there is voice). The data packets are sent in a non-reserved fashion. Each data is sent individually without any reservation.

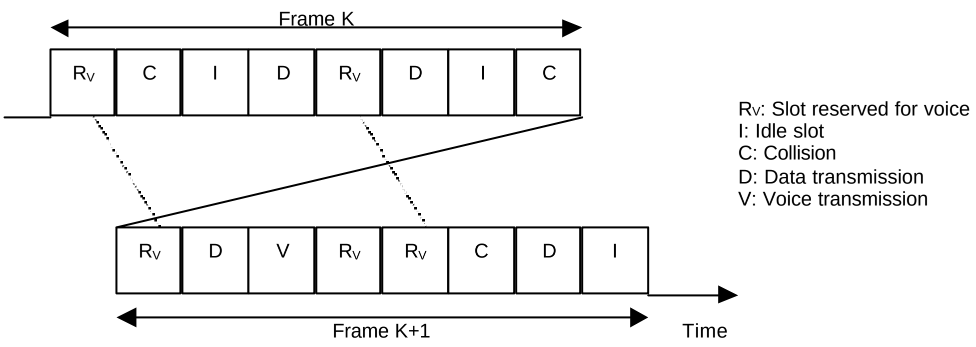

Figure 3.8: Packet reservation multiple access protocol.

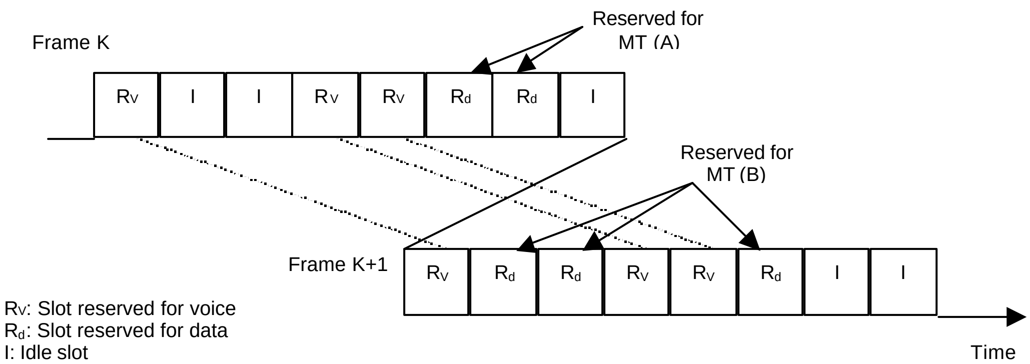

Figure 3.9: Integrated packet reservation multiple access protocol.

Many random reservation protocols are extensions of PRMA. Integrated Packet Reservation Multiple Access (IPRMA) [Wong93] (Figure 3.9) is a PRMA protocol that provides reservation for data traffic. IPRMA follows the rules of PRMA for voice traffic transmission and puts a limitation on the number of slots that can be reserved. To provide fewer collisions and higher throughput, IPRMA allows data traffic to reserve slots horizontally. Voice traffic reserves slots vertically in that the same slots of the following frames are reserved. Data traffic reserves the remaining empty slots. The slots that can be reserved by data traffic are fewer than those reserved by voice traffic.

3.2.3.4 Demand Assignment MAC Protocols

Demand assignment protocols combine random access and guaranteed access protocols, and allocate bandwidth to the MTs according to their QoS requirement. Networks such as wireless ATM networks and wireless voice communication networks require guaranteed QoS for their multimedia traffic. To achieve this, the demand assignment protocols first gather the information on the requests of the MTs. Then they schedule sufficient bandwidth to satisfy the needs of the MTs. The information gathering usually takes place through the reservation phase in the uplink stream. The reservation phase usually consists of many contention mini-slots. These slots are used by the MTs to transmit their requests. This is the only part of the demand assignment protocol that involves contention. The mini-slots are used instead of larger data slots to minimize the wastage of the bandwidth due to collisions.

Once the BS has information on the requests of the MTs, the BS uses a scheduling protocol to assign data slots to the MTs. The assignments of the uplink data slots are notified through the downlink stream. The data for the MTs from the BS are usually transmitted after the notifications. The last phase of the protocols is the data transmission phase. The MTs transmit their data without any collision in the data slots assigned by the BS.

In demand assignment protocols, the BS is required to do a large amount of processing. It must define the structure of both the uplink and the downlink streams, usually in a frame structure, and also performs the scheduling of data slots, acknowledgement of requests, and time synchronization.

Time Division Multiple Access with Dynamic Reservation (TDMA/DR)

The structure of the TDMA/DR protocol [Wong00] (Figure 3.10) is based on TDMA frames. Only the uplink frame was mentioned by the authors. The uplink frame can be divided into two phases: the reservation phase and the data transmission phase. In the reservation phase, a large number of mini-slots are used for requesting bandwidth in the data transmission phase of the next uplink frame. The traffic generated by MTs are grouped into three classes: class 1 - real-time VBR and time-critical data traffic, class 2 - CBR voice with burst switching (on/off), and class 3 - non time-critical data.

The reservation phase is divided into two parts. The first is called available request channel. It is used to send requests randomly by all classes based on the slotted-Aloha access protocol. The second is called the used request channel. It is reserved for class 1 traffic that has successfully transmitted their requests through the available request channel. The reserved used request mini-slots allow waken class 1 traffic to quickly send their requests without collisions, and hence decrease the delay of real-time VBR traffic. The protocol does not see voice traffic as real-time VBR traffic but as short CBR traffic. The voice traffic (class 2) must requests data slots through the available request channel every time a talk spurt occurs. A slot in the data period is dedicated to the voice traffic until the talk spurt ends. No advantages are given to the voice traffic. When class 1 and class 2 traffic have collisions in the request channel, they retransmit their request immediately in the following frame. Class 3 traffic back off when similar situation arises.

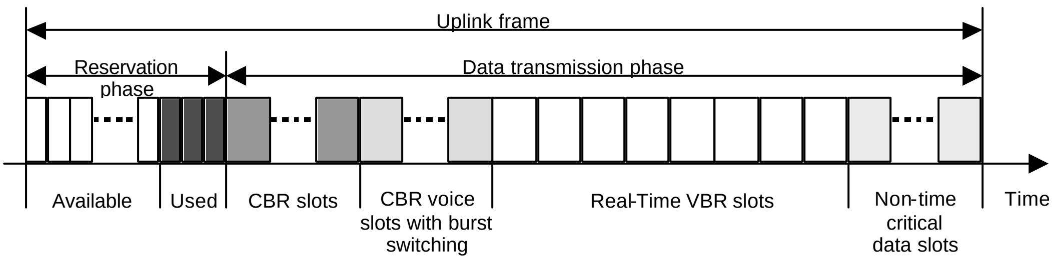

Figure 3.10: Time division multiple access with dynamic reservation protocol.

The data transmission phase is divided into four sections: CBR, CBR with burst switching, real-time VBR, and non-time-critical data. Each section contains a different number of data slots that are assigned by the BS. The boundaries between each section are movable depending on the traffic load. The protocol wastes bandwidth when a used request mini-slot is assigned to VBR traffic, since the VBR traffic only requires the mini-slot for a short period of time.

Dynamic Hybrid Partitioning (DHP)

The DHP protocol [Rezv99] (Figure 3.11) operates in a TDD environment. It is designed particularly to deal with the idle-VBR problem. A single frame with an uplink and a downlink period is proposed. The uplink period is further divided into two phases: the reservation phase and the data transmission phase. The reservation phase consists of two types of request mini-slots. The contention mini-slots are random access slots and are based on the slotted-Aloha access protocol. They are available for all MTs to transmit their requests of bandwidth in the data transmission phase. The second type of mini-slots are used for the MTs to transmit their waken idle-VBR requests. These mini-slots are assigned to VBR traffic when they become idle. Each mini-slot is assigned to a particular VBR traffic. This allows the VBR traffic to transmit their request without going through any contention, and hence minimizes the delay. A mini-slot is released when the idle-VBR traffic that was assigned to it no long idles. If the mini-slot is not assigned to other idle-VBR traffic, it is changed to a contention mini-slot. The data transmission phase of the uplink period is divided into three partitions for different ATM traffic: CBR (Constant Bit Rate traffic), VBR (Variable Bit Rate traffic), and UBR (Unspecified Bit Rate traffic). Each partition is dedicated to a type of traffic with a movable boundary.

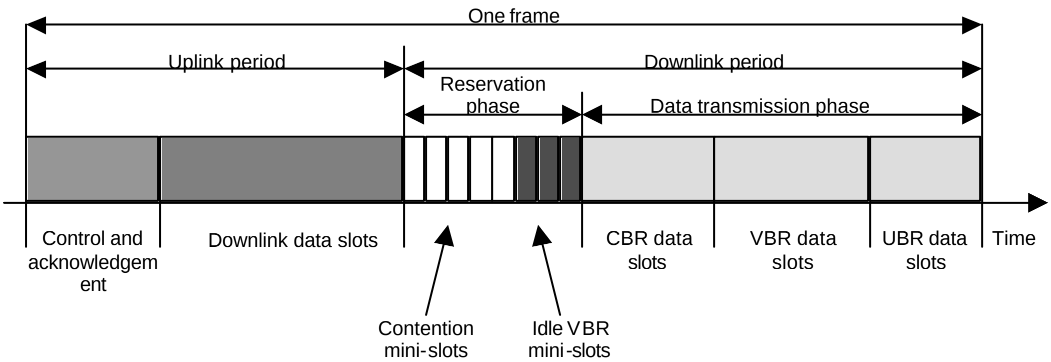

Figure 3.11: Dynamic hybrid partitioning protocol.

The downlink period is made up of two parts. Part one includes all required information such as modem preamble, header, control information, and acknowledgements for the previous uplink period. The second part contains data from the BS.

The approach of the DHP protocol in mini-slot reservation is more efficient than that of the TDMA/DR protocol. The DHP protocol reserves a mini-slot only in the idle-mode of the VBR traffic, whereas for the TDMA/DR protocol, a mini-slot is assigned to the VBR traffic for the entire duration of its connection.

Fair Access Fair Scheduling (FAFS)

The FAFS protocol [Jain99] (Figure 3.12) is a protocol based on TDD. A fixed length frame is used, which can be divided into four phases: synchronization, reservation, acknowledgement, and data transmission. The synchronization phase and acknowledgement phase are in the downlink direction. The synchronization phase is used to transmit the information about the field boundary and to synchronize the MTs with the BS. The acknowledgement phase transmits the information about the data slots allocation of the data transmission phase and acknowledges transmissions from the previous frame. The reservation phase is in the uplink direction.

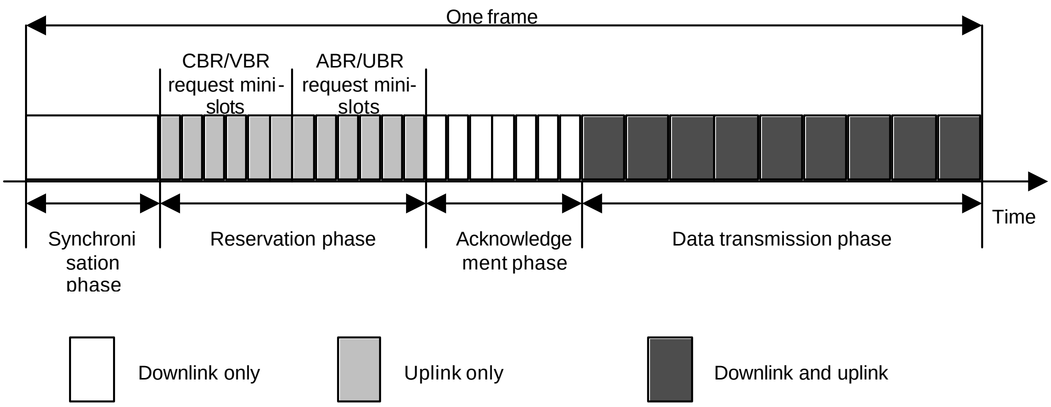

Figure 3.12: Fair access fair scheduling protocol.

The reservation phase is divided into two partitions. They both contain many request mini-slots. The slotted-Aloha access protocol is used by the MTs to access these contention mini-slots. CBR and VBR traffic transmit their requests through part one. ABR and UBR traffic transmit their requests through part two. The reservation phase partition can eliminate the contentions between CBR/VBR and ABR/UBR traffic, and hence decrease the delay of CBR and VBR traffic. Eight bits of information about the queue condition and the number of failed attempts of an MT are added to the request packets of ABR and UBR traffic. This allows the BS to schedule according to their conditions. The data transmission phase is used to transmit the downlink and uplink data packets.

Adaptive Framed Pseudo-Bayesian Aloha (AFPBA)

The AFPBA protocol [Haba00] (Figure 3.13) is focused on the reservation phase of the demand assignment protocols. The mini-slots in the reservation phase are assigned to different classes of traffic. In the case of a wireless ATM network, at least four classes of traffic exist. Each class sends their requests through the mini-slots that are assigned to them. The number of mini-slots assigned to each class is dynamic, and is based on an adaptive slot assignment algorithm. The algorithm uses the Poisson arrival rates and the number of backlogged packets of each class to estimate the number of contention mini-slots required for each class.

By assigning a large number of mini-slots to a class, the delay of that class can be decreased. This can improve the delays in sending requests.

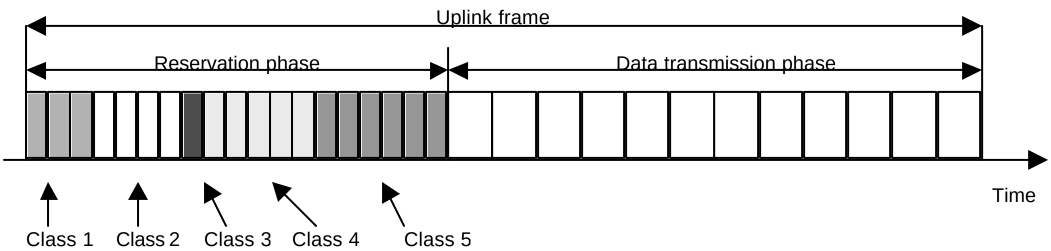

Figure 3.13: Adaptive framed pseudo-Bayesian aloha protocol.

Sequence Number based Dynamic MAC (SND-MAC)

The SND-MAC protocol [Zhij00] (Figure 3.14) uses a frame structure based on TDD. A frame consists of an uplink and a downlink frame. This protocol has two operating modes: Heavy Overload Mode (HOM) and Light Overload Mode (LOM). The structure of the protocol changes when the operating mode changes. HOM is engaged when the load on the uplink stream is high. Once the average load of the uplink stream drops to a certain level, the protocol switches the operating mode to LOM. Each mode has its own uplink and downlink frame structure.

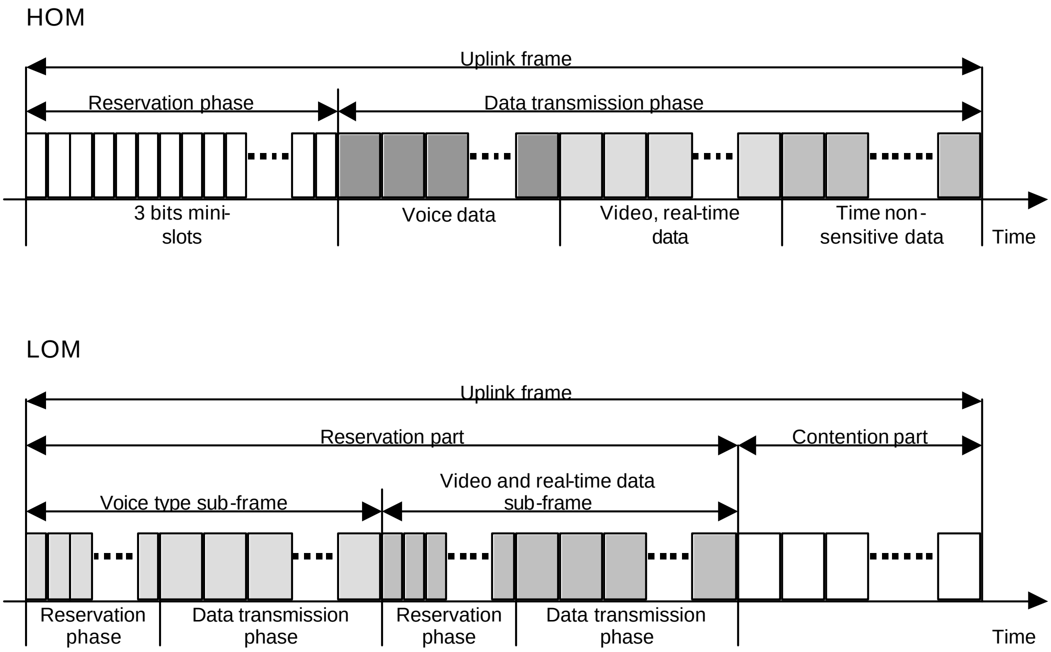

Figure 3.14: Sequence number based dynamic MAC protocol.

In HOM, the protocol is a deterministic guaranteed access protocol. The reservation phase is divided into many 3-bit mini-slots. The number of mini-slots corresponds to the number of MTs in the system. Each MT is assigned a mini-slot. The reservation phase is contentionless, since each MT has a dedicated request slot. When an MT wants to request some resources, it transmits a 3-bit message in its dedicated request mini-slot. This allows an MT to request up to five data slots. The aim of HOM is to eliminate the large number of collisions caused by random access in heavy traffic. If all of the MTs transmit their requests together in the same uplink frame, then the channel is fully utilized and there is no wastage due to collisions.

The downlink frames of HOM are similar to its uplink frames. They follow immediately after an uplink frame. The 3-bit downlink mini-slots are used to acknowledge the uplink request mini-slots and assign the data slots in the next uplink frame. The downlink data slots follow the acknowledgement mini-slots.

In LOM, the uplink frame is divided into two parts: reservation and contention. The contention part contains many free data slots. ABR and UBR traffic transmit their data using the slotted-Aloha access protocol without reservations in these data slots. CBR and VBR traffic use the reservation part of the frame. The reservation part is further separated into two sub-frames, one for voice traffic and the other for video and real-time data traffic. Each sub-frame is a mini demand assignment frame. The size of each frame is fixed, with a reservation phase and a data transmission phase. The reservation phase consists of contention mini-slots for requesting data slots in the data transmission phase of the sub-frame in the next uplink frame. The data slots are assigned by the BS to the MTs. The LOM downlink frames are similar to the HOM downlink frames. An LOM downlink frame has extra control information fitted in the header of the frame.

The use of random access in a heavily loaded environment can cause many collisions. HOM avoids collisions by pre-assigning a mini-slot to each MT. This approach is not scalable when a large number of MTs is presented in the system. Many non-transmitting mini-slots are wasted if the number of requesting MTs is low.

Multiservices Dynamic Reservation (MDR)

In MDR [Rayc94] (Figure 3.15), the uplink stream is divided into frames. Each frame consists of a reservation phase and a data transmission phase. The reservation phase is made up of many mini-slots. All MTs must request data slots from the BS before they transmit their data. The MTs send request packets through the mini-slots using the slotted-Aloha protocol. The mini-slots can minimize the bandwidth wastage due to collision. When a collision occurs, only a few bytes are wasted. After a successful transmission in the reservation phase, an MT listens to the downlink frames for the reply from the BS. If there are available data slots in the uplink data transmission phase, the BS assigns the available data slots using a scheduling algorithm.

In the data transmission phase, the data slots are divided into two types. Type 1 is dedicated to CBR traffic. Type 2 is for VBR and other data traffic. The MDR protocol gives the highest priority to CBR traffic; therefore, it does not provide guarantees for VBR traffic. Once type 2 slots are all occupied, the VBR traffic cannot request any more resources even if there are still available data slots of type 1 in the reservation phase.

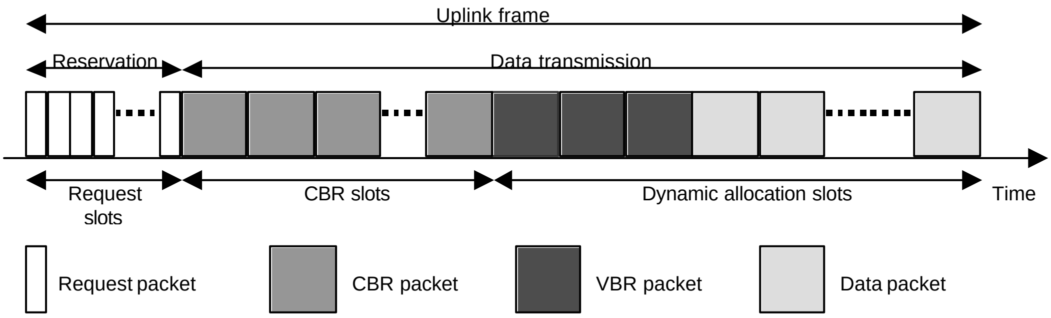

Figure 3.15: Multiservices dynamic reservation protocol.

Dynamic TDMA with Piggybacked Reservation (DTDMA/PR)

The DTDMA/PR protocol [Qiu96] (Figure 3.16) is a TDD protocol. The uplink frame consists of two phases: reservation and data transmission. The protocol uses mini-slots to request bandwidth in the reservation phase of the uplink frames. Once the request packets have transmitted successfully, the BS assigns data slots in the data transmission phase according to the priority of the different traffic types. The data transmission phase can be divided into two parts: long-term reservation and short-term reservation. Long-term traffic such as CBR and VBR are assigned to the long-term reservation part. The other data traffic is assigned to the short-term reservation part.

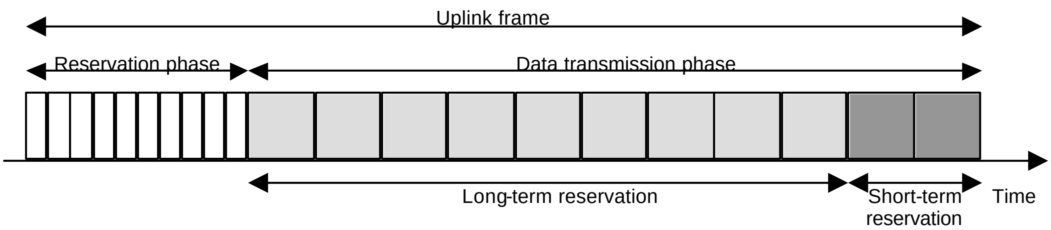

Figure 3.16: Dynamic TDMA with piggybacked reservation protocol.

The boundary between the two parts is movable. A technique called piggybacking is employed in DTDMA/PR for VBR traffic. VBR traffic usually comes in bursts. If data slots are reserved in each frame as in CBR traffic, the data slots are wasted when VBR traffic is in the idle mode. To overcome this, every time VBR traffic has something to send, it requests the number of data slots it needs through the reservation phase. If there is extra VBR traffic generated while the MTs are transmitting their traffic, the new traffic has to request data slots through the next reservation phase with a possibility of collision. Piggybacking is designed to fix this delay. A message is fitted at the end of the data packet. The message is used to request more data slots without going through the contention reservation procedures.

Adaptive Request Channel Multiple Access (ARCMA)

ARCMA [Chew99] (Figure 3.17) is a frameless demand assignment protocol. The slots are assigned individually without any structure. The protocol is based on TDD. Two phases exist in the uplink stream: the reservation phase and the data transmission phase. Similar to other demand assignment protocols, the reservation phase consists of mini-slots and the data transmission phase consists of data slots. The number and position of these slots are assigned by the BS alone. Since there are no frame structures, the MTs are required to monitor the downlink stream at all times. The advantage of a frameless structure is that the BS can assign resources to different requests. If a large number of collisions occur in the reservation phase, the BS can allocate more mini-slots to relieve the traffic congestion. In the framed approach, some data slots in the data transmission phase can be unused when the reservation phase is heavily congested.

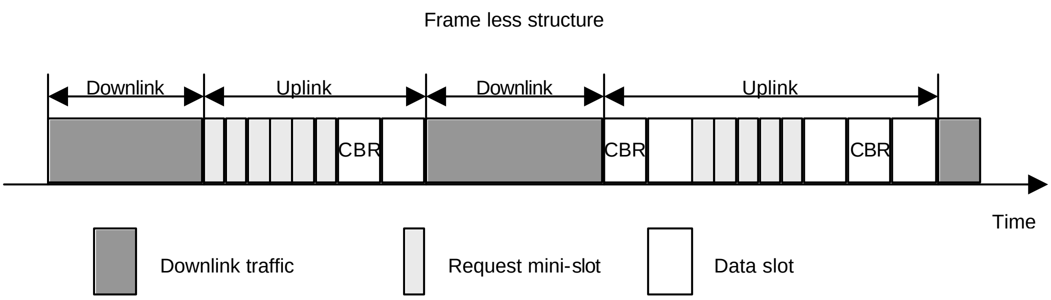

Figure 3.17: Adaptive request channel multiple access protocol.

The QoS requirements for CBR and VBR traffic can be easily matched in this protocol. The BS can assign the correct number of data slots for the traffic without restriction. The downside of the protocol is the large overhead required in the downlink stream to inform the MTs when to transmit.

TDD System for WATM Network (TWATM)

The MAC protocol presented in TWATM [Le99] (Figure 3.18) is based on TDD. A frame is divided into a downlink sub-frame and an uplink sub-frame. The downlink sub-frame can be separated into three periods: the initial broadcast period, the downlink data period, and the post-broadcast period. The function of the initial broadcast period is to allow synchronization of the frame and to outline the frame structure. The post-broadcast period is used to acknowledge the transmissions from the last uplink sub-frame and to assign the data slots in the data transmission phase of the uplink sub-frame.

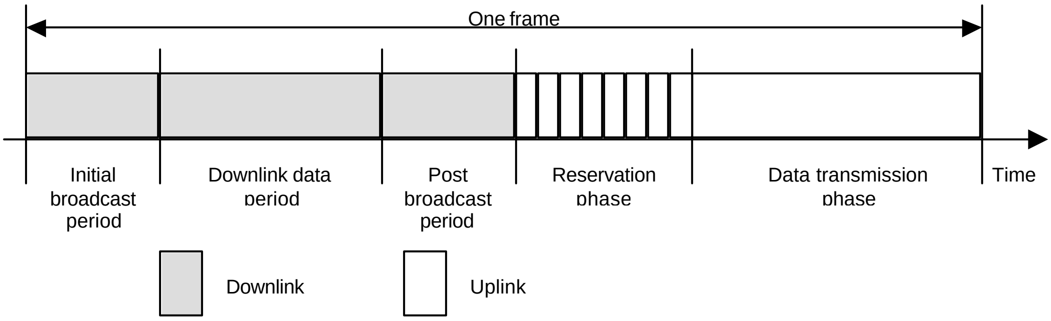

Figure 3.18: TDD system for WATM network.

Like other demand assignment protocols, the uplink sub-frame has a reservation phase and a data transmission phase. The reservation phase consists of contention mini-slots based on the slotted-Aloha access protocol. The data transmission phase consists of data slots that can be assigned by the BS to the different MTs. TWATM has a special feature that is not seen in other demand assignment protocols. It allows an MT to transmit two request packets (the same packet) in two different mini-slots in the reservation phase of an uplink frame when the traffic load is low. This “double requests” strategy is used to increase the probability of success in the mini-slot contention. When the load in the reservation phase become heavy, the MTs switch to a normal mode, in which an MT can only send one request in a frame. The double requests sending can greatly decrease the throughput of the reservation phase if it is used when the traffic load is high.

Distributed-Queuing Request Update multiple Access (DQRUMA)

DQRUMA [Karo95] (Figure 3.19) uses FDD for the uplink and downlink transmissions. The frames used in the protocol are very small and fixed in size. A standard uplink frame consists of a request mini-slot (the reservation phase), a piggybacking field, and a data slot (the data transmission phase). Unlike other demand assignment protocols, a DQRUMA frame can only carry one data packet. The request mini-slot can be accessed using the slotted-Aloha access protocol. The piggybacking field is for the MT that is assigned to the data slot. If the MT has more data packets to transmit, it can request more data slots using the piggybacking field.

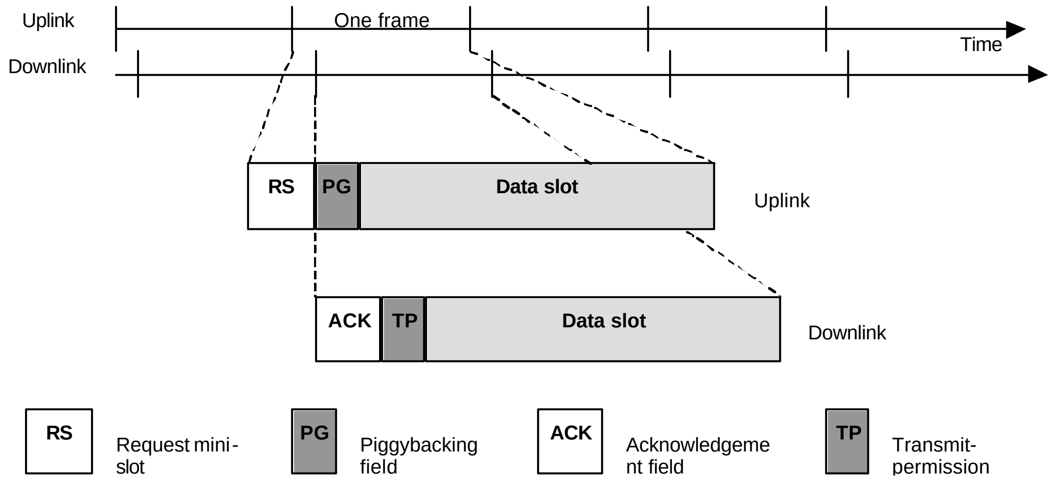

Figure 3.19: Distributed-queuing request update multiple access protocol.

A standard downlink frame is made up of an acknowledgement field, a transmit-permission field, and a downlink data slot. The acknowledgement field provides information on the request mini-slot of the uplink frame. The uplink frames and the downlink frames are arranged in FDD so that the acknowledgement of a request in the downlink frame follows immediately after the transmission of the request in the uplink frame. An MT is acknowledged on the result of its request transmission before the next uplink frame begins. The transmit-permission field contains the ID of an MT that will be transmitting data in the data slot of the next uplink frame. The downlink data slot simply carries the downlink data traffic from the BS to a particular MT.

The uplink frame and the downlink frame are converted to multiple request mini-slots and multiple acknowledgement fields (Figure 3.20) when both the uplink and the downlink data slots are idle. The uplink frame is divided into N number of mini-slots with N-1 request mini-slots. The downlink frame is divided into N number of mini-slots with N-1 acknowledgement fields and one transmit-permission field. The conversion of a data slot into many request mini-slots can increase the accessibility of the bandwidth and reduce the waste of resources.

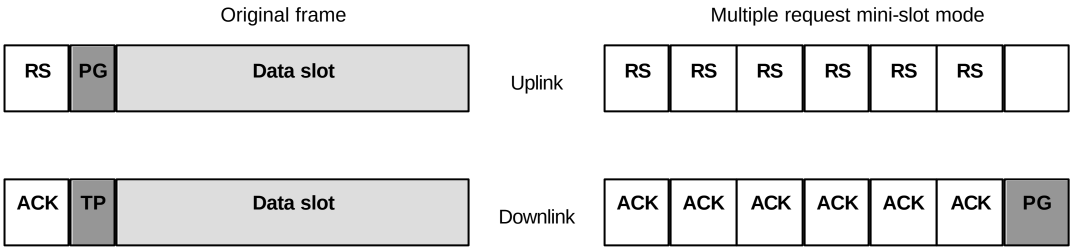

Figure 3.20: Frame conversion in DQRUMA.

When an MT has data to transmit, it first transmits a request packet through a request mini-slot in an uplink frame based on the slotted-Aloha access protocol. After the MT transmits the request packet, it checks the acknowledgement field in the downlink frame immediately for the result of its request. If there a collision has occurred in the uplink request mini-slot, the MT retransmits the request packet according to the random access protocol. If there is no collision, the MT listens to the transmit-permission field for more instructions. When the MT hears its ID in the transmit-permission of the downlink frame, it knows the data slot of the next uplink belongs to it. The MT transmits its data packet in the next uplink frame. If the MT has more data packets to transmit, it sends a message in the piggybacking field to request more data slots.

Mobile Access Scheme based on Contention and Reservation for ATM (MASCARA)

MASCARA [Mikk98] (Figure 3.21) is the MAC protocol for the Magic WAND project. It forms the basis for the HIPERLAN type 2 standardization. MASCARA is a TDD based MAC protocol. It uses variable-length frames. A MASCARA frame is made up of three phases: a broadcast phase, a data transmission phase, and a reservation phase. The boundary of each phase is movable and is controlled by the BS. The broadcast phase is in the downlink direction. It is used to notify all MTs of the structure of the current time frame and the scheduled uplink transmissions and to acknowledge the requests from the previous frame. The data transmission phase consists of a downlink data phase and an uplink data phase. The downlink data phase carries data packets from the BS to the MTs. The uplink data phase contains many data slots. The length of each data slot is variable. Each data slot is assigned to an MT. The MTs request these data slots by sending their request through the reservation phase. The reservation phase consists of many request mini-slots. It is used by the MTs to send their requests or sometimes their control information. All packets that transmit through this phase use the slotted-Aloha access protocol. A request packet requires one mini-slot, and a control information packet requires two mini-slots. After the request reception, the BS makes uplink data slot assignments based on a leaky bucket token scheme called Prioritized Regulated Allocation Delay-oriented Scheduling (PRADOS) [Colo99].

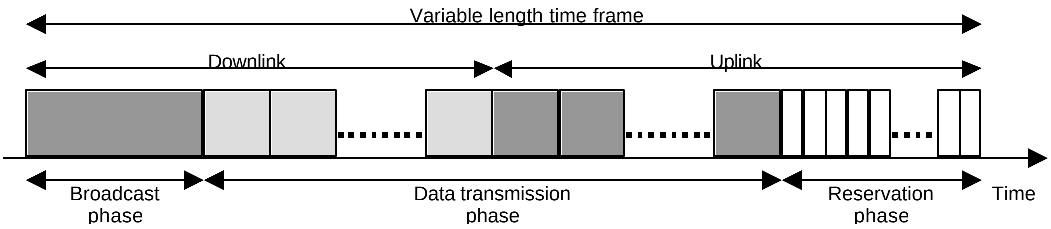

Figure 3.21: Mobile access scheme based on contention and reservation for ATM.

Dynamic Slot Assignment ++ (DSA++)

DSA++ [Petr95] (Figure 3.22) is an FDD based MAC protocol. Both the uplink and the downlink frame consist of many slots. The number of slots in a frame is fixed. An uplink frame contains the same number of slots as a downlink frame. The size of each slot is just large enough to carry an ATM cell. A slot in an uplink frame can be either a data slot (data transmission phase) or a random access channel (reservation phase containing many request mini-slots). The BS defines the type of each slot through the signalling channel of the downlink frame. Other than number of slots, an uplink frame does not have a specific structure.

The downlink frame is made up of a signalling channel and many data slots. A signalling channel is the same length as other slots. The function of the signalling channel is to announce the structure of the next uplink and downlink frames. It also serves to reserve data slots of the next uplink frame and acknowledge transmissions in the random access channel. When a slot in an uplink frame is assigned to be a random access channel, the slot is divided into many request mini-slots. The MTs with something to send can access the random access channel using the slotted-Aloha access protocol. The MTs are only required to listen to the signalling channel for the upcoming frame format and for information about their transmissions in the random access channel. This means an MT can sleep when it has nothing to receive and transmit. A great amount of power can be saved.

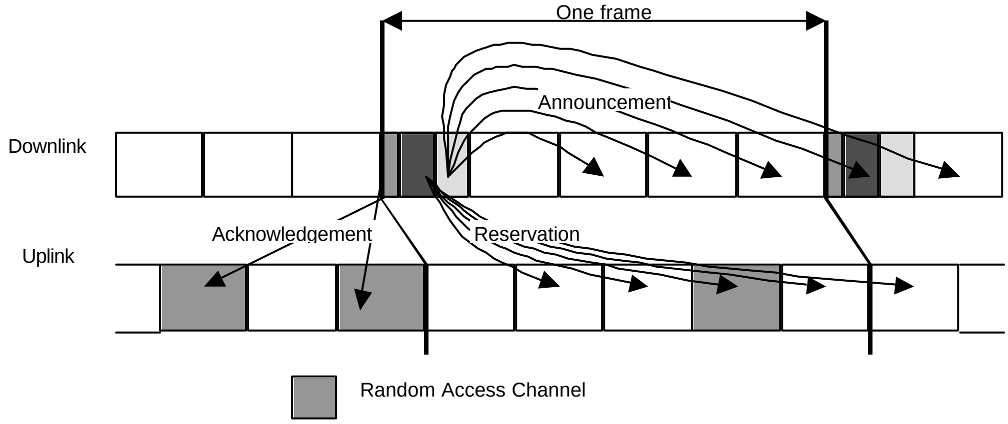

Figure 3.22: Dynamic slot assignment ++ protocol.

Collision Based Reservation Multiple Access (CBRMA)

CBRMA [Jian98] (Figure 3.23) is a TDD based protocol. The frequency stream is divided into time frames. Each frame has four components: a downlink broadcast phase, a downlink data transmission phase, a reservation phase, and a uplink data transmission phase. The broadcast phase helps the MTs to synchronize with the BS. It carries the information about the frame structure and the acknowledgements of the previous requests. The downlink broadcast phase is responsible for carrying data from the BS to the MTs.

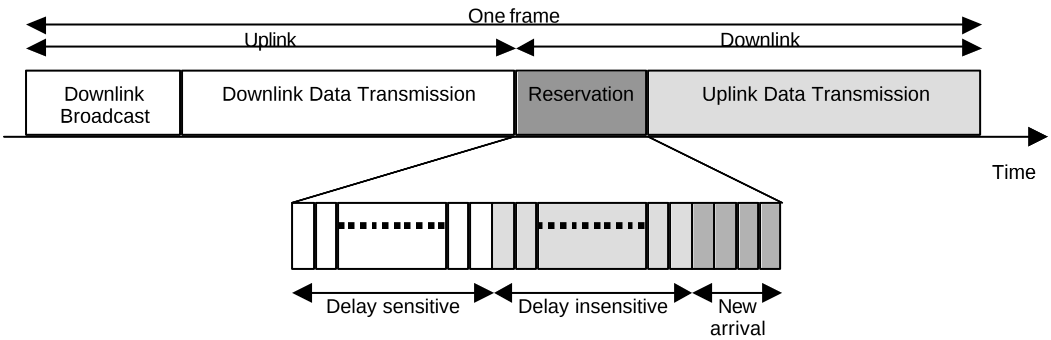

Figure 3.23: Collision based reservation multiple access protocol.

The reservation phase consists of many request mini-slots. These mini-slots are grouped into three groups, which create three different periods in the reservation phase. Each period is for a different type of traffic. The first period is for delay-sensitive traffic (such as VBR), the second is for delay-insensitive traffic, and the third is for new arrivals. The number of mini-slots in each period is changeable. A newly arrived MT transmits its request packet in the new arrival period. All other traffic uses the delay-insensitive traffic period to request bandwidth. The delay-sensitive traffic period is not for random access as in the other two periods. When delay-sensitive traffic becomes idle, it is assigned a mini-slot in this period. Each slot can be assigned to two MTs with the delay-sensitive traffic that is currently idle. When an idle MT with delay-sensitive traffic awakes from the idle state, it can request data slots through the pre-assigned mini-slot without contention. Because, at a maximum, two MTs can be assigned to a reserved mini-slot, a collision is possible when both MTs transmit their request packet at the same time. Since the mini-slot is assigned to known MTs, the BS assumes that both MTs have transmitted their request packet at the same time when a collision occurs. The BS can then assign data slots to the two MTs. It is possible that when the capture effect occurs, the request packet from one of the MTs cannot be received.

Once the transmission of a request from an MT is successful, the BS assigns data slots in the uplink data transmission phase to the MT and acknowledges the MT through the downlink broadcast phase. The assignment of the data slots is accorded to a scheduling algorithm. If the MT has more data packets to transmit, it can send its request using piggybacking.

Dynamic Packet Reservation Multiple Access (D-PRMA)

D-PRMA [Alas99] (Figure 3.24) has evolved from PRMA (a random reservation protocol). Because it uses request mini-slots for reserving data slots for voice traffic, we classify it as a demand assignment protocol. The uplink stream and the downlink stream are divided into frames. The author did not specify whether the protocol is a TDD based protocol or an FDD based protocol, or describe the structure of a downlink frame. The focus is on the uplink frame structure. The uplink frame is made up of three phases: a voice data transmission phase, a data phase, and a voice reservation phase. The voice data transmission phase consist of data slots that are dedicated to voice traffic. To access these slots, an MT has to send a request packet through the voice reservation phase. The voice reservation phase is the equivalent of the reservation phase in other demand assignment protocols. The phase consists of mini-slots. The BS assigns voice data slots to the MT according to a scheduling algorithm. This procedure is for voice traffic and constant bit rate traffic only. Other types of traffic send their data packets in the data phase. The access of the data phase is based on contention. The MTs use the slotted-Aloha access protocol to send their data packets in this phase. No reservation is used.

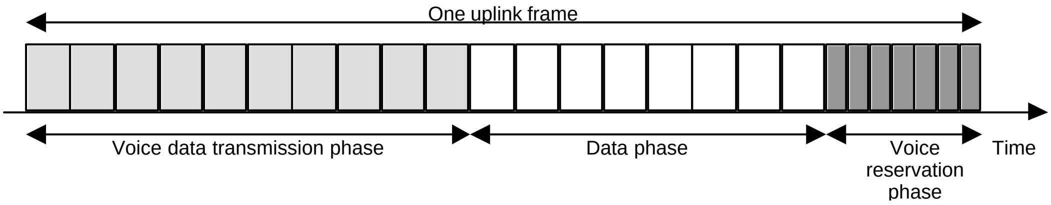

Figure 3.24: Dynamic packet reservation multiple access protocol.

Collision Resolution and Dynamic Allocation (CRDA)

CRDA [Lenz01] (Figure 3.25) is an improved version of PRMA. Unlike its ancestor, CRDA is not a random reservation protocol. It demands resources through a request channel with a non-data carrying request packet. FFD is used as the duplex mode. The uplink and downlink streams are divided into frames. Each frame contains N number of slots. Two types of slots are available in the uplink frame: request slots and data slots. The request slots represent the reservation phase and the data slots represent the data transmission phase. The uplink frame begins with the reservation phase and is followed by the data transmission phase. The request slots are the same size as the data slots.

The downlink frame consists of acknowledgement slots and data slots. The downlink frame is arranged so that it is off-set by the number of request slots in the uplink frame. This allows the requests from the uplink to be acknowledged immediately when the uplink reservation phase is completed. The downlink data slots carry not only the downlink data, but also information on the uplink data slot assignment.

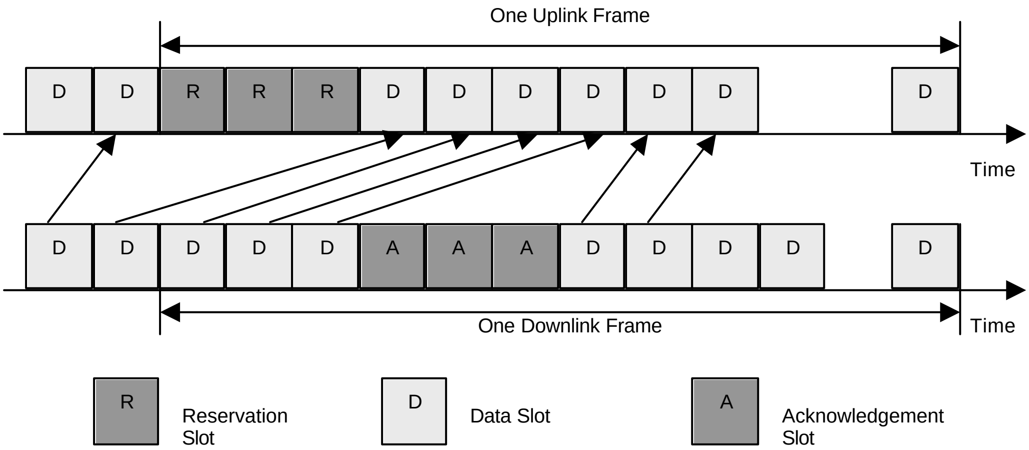

Figure 3.25: Collision resolution dynamic allocation protocol.

When an MT has something to transmit, it first sends a request packet in the reservation phase. If the transmission is successful, it listens to the downlink data slots for the uplink data slot assignment. The result of the transmission is reported in the acknowledgement slots of the next downlink frame. Further requests can be obtained by using piggybacking. CRDA supports only voice and data traffic.

3.2.4 Wireless Medium Access Control Protocols for Ad-hoc Centralized Networks

The ad-hoc centralized MAC protocols are designed for networks based on an ad-hoc centralized topology. This topology is a combination of ad-hoc topology and the centralized topology. A network is generally viewed as an ad-hoc network when it can be constructed in any place without a stationary base station. In an ad-hoc centralized network, there is a centralized administrator (a mobile base station) in the network. The communication is centralized. The following is an example of an ad-hoc centralized MAC protocol.

Bluetooth Radio System

The Bluetooth radio system [Haar00] is a low cost indoor network. It is ad-hoc in nature. Bluetooth is based on FH-CDMA. It uses the 2.45GHz ISM band (Industrial, Scientific, and Medical band) and has 79 hop carrier channels. The fullduplex communication is achieved by applying TDD. A randomly selected hop carrier channel is used by a Bluetooth network. (Also called a Bluetooth piconet. It is a short distance network with a diameter of 10 meters.) A Bluetooth piconet consists of eight or fewer wireless nodes. One of the nodes in the piconet is a master node and the rest are slave nodes. The master and slave relationship exists only when the piconet exists. The initiator of the piconet becomes the master. All Bluetooth nodes have the same physical capability and can become master nodes or slave nodes. The master node in a piconet acts as the BS in the centralized topology and the slave node acts as the MT. The topology of the piconet is the same as a centralized topology except the BS (master) is mobile not stationary. The slaves can only communicate with the master and not with the other slaves. A polling based guaranteed access MAC protocol is used in Bluetooth.

Figure 3.26 shows the MAC protocol used in Bluetooth. The master polls each slave in a round robin fashion. The master controls the access of the medium. The master polls the slave A for data with a downlink poll packet. After hearing the poll, slave A immediately transmits its packet. The master then begins a downlink transmission. This downlink transmission contains two packets and is for slave E. Slave E acknowledges the master immediately after receiving the two packets. The master goes on and polls the next slave on its list. Slave B is polled. Since slave B does not have a data packet to send, it replies the master with a nothing-to-send packet. The master controls the entire traffic flow. The number of packets that can be transmitted by the slaves in each poll is controlled by the scheduling algorithm of the master node.

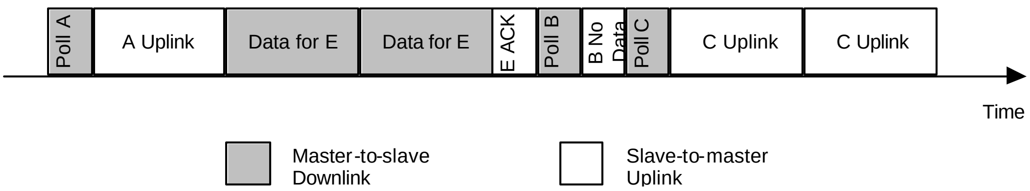

Figure 3.26: MAC protocol of Bluetooth radio system.

3.3 Summary

The wireless medium access control protocols can be classified into many different classes according to their topology and the approach they use. The MAC protocols designed for ad-hoc networks are very different from those designed for centralized networks. Most of the MAC protocols for the ad-hoc networks use handshakes and busy tones to avoid collisions. The performances of these protocols are not as good as the MAC protocols for centralized networks. This is mainly due to the propagation delays generated by the handshakes while trying to organize collision free transmissions. The MAC protocols designed for centralized networks use the BS as the central administrator to effectively allocate bandwidth for the MTs. Among them, the demand assignment protocols have the best performance. In the next chapter, we present an algorithm that can be added to some of the demand assignment protocols to improve their performance.

Chapter 4

An Improved Channel Reservation Scheme for Demand Assignment Medium Access Control

Many strategies have been proposed to improve the performance of the demand assignment MAC protocols. In this chapter, we introduce a channel reservation scheme based on a new concept called transmission probability assignment. This protocol is an add-on solution to demand assignment MAC protocol. We first look at channel reservation strategies that have already been used in demand assignment MAC protocols. Then we introduce the concept of transmission probability assignment and the transmission probability based dynamic slot assignment (TRAPDYS) protocol.

4.1 Strategies Used to Improve Demand Assignment MAC Protocols

Demand assignment MAC protocols have been used in several standardized wireless networks, including third generation wireless telecommunication networks. The demand assignment MAC protocols can effectively utilize wireless bandwidth of radio channels and can support a large number of MTs. Although demand assignment

MAC protocols can produce good results, further performance improvement can be obtained by using the following additional strategies of channel reservation. These protocols are described in detail in Chapter 3:

A. **Request Slot Class Assignment (RSCA)** [Jain99, Haba00, Zhij00, Jian98]: The request slots in the reservation phase are grouped into classes. MT of a given class can only transmit its request packets within a particular set of request slots. Usually the high priority traffic classes are assigned with more request slots. Increasing the number of request slots decreases the probability of collisions. This allows the high priority traffic classes to transmit their request packets with shorter delay, and improves the chance of providing the requested QoS requirements.

B. **Request Slot Pre-assignment** [Wong00, Rezv99, Zhij00]: In this method, some or all of the request slots in the reservation phase are pre-assigned to specific MTs for a period of time. The purpose of these pre-assigned slots is to allow the assigned MTs to transmit their request packets without interruption or collision. They can only be accessed by the assigned MTs. Each request slot is reserved for only one MT. In DHP [Rezv99] and TDMA/DR [Wong00], the pre-assigned slots are reserved for these MTs with VBR traffic that are currently in the idle mode. In the HOM of SND-MAC [Zhij00], all request slots are pre-assigned. Each MT has its own dedicated request slot and no collision can occur in the reservation phase. The elimination of collisions can produce high throughput and allows the requests from the MTs to reach the BS quickly. However, if the MTs owning the pre-assigned slots have nothing to transmit, the pre-assigned slots are wasted.

C. **Data packet transmission in the reservation phase** [Mikk98]: A request slot in the reservation phase is usually several bytes in length. By combining two or more request slots, a time slot that is capable of transmitting a data packet can be constructed. In MASCARA [Mikk98], an MT can transmit its control information packet (such as information concerning the current network conditions) or data packet either during the reservation phase or during the data transmission phase. In a light traffic load, the MTs are encouraged to transmit their control information packets in the reservation phase. This is to further utilize the unused request slots and decrease the delay caused by a given reservation procedure.

D. **Double request transmission** [Le99]: The transmission of double requests is used to increase the probability of successful reservation. In TWATM [Le99], when the traffic load is low, an MT is allowed to transmit a request packet twice in different request slots of the same reservation phase. This is to double the chance of successful transmission. If both requests get through without collision, the BS will only listen to one request. This method is only suitable in a low traffic load situation. If the traffic load is high, the double transmission method can cause many collisions and long delays. Thus, this strategy should be then switched off.

E. **Request scheduling**: The efficiency of a demand assignment MAC protocol depends heavily on the way in which data slots are scheduled by BS after receiving requests from MTs. In [Wong00, Rezv99, Zhij00, Rayc94, Qiu96], the BS schedules requests depending on the number of free data slots available in a given type of traffic. The uplink data transmission phase is divided into groups of data slots for different types of traffic. Once the data slots of a given traffic class have all been assigned, the traffic of the same class must wait until the next frame before more bandwidth is assigned to this class, even if there are unused data slots in other traffic classes. Such scheduling method can cause the bandwidth to be under-utilised and result in a poor QoS. Other protocols use more sophisticated scheduling methods [Jain99, Haba00, Chew99, Le99, Karo95, Mikk98], which can put heavy processing loads on the BS. The BS assigns each data slot according to a specific scheduling method and it is not limited by the frame structure. The scheduling methods usually satisfy the needs of the high priority traffic first before they give data slots away to non-critical traffic.

F. **Frameless structure** [Chew99]: Most demand assignment MAC protocols have a basic frame structure. An uplink frame is divided into a reservation phase and data transmission phase. Many framed demand assignment MAC protocols allow variable frame lengths and variable phase lengths, but phases always come in the same order. A frameless protocol has been proposed in [Chew99]. In this protocol, each slot (request slot and data slot) is defined by the BS. The MTs are required to monitor the downlink stream at all times. The traffic with a higher QoS requirement requires a faster response from the BS. The protocol satisfies such a requirement by creating more data slots for that traffic. A frameless structure allows freer assignment of data channel to active MTs.

G. **Piggybacking** [Qiu96, Karo95, Lenz01]: Requests are inserted into uplink data packets and transmitted during the data transmission phase. This request information allows the MT to request more uplink data slots without going through the reservation phase. Requests can reach the BS faster because there are no collisions. Hence, this scheme provides better performance.

Many of these strategies can be implemented together in one demand assignment protocol to further increase the efficiency of the protocol.

4.2 Dynamic Random Channel Reservation

Prioritised access with short delays is preferred for multimedia traffic. Multimedia applications such as voice or video generate Variable Bit Rate (VBR) traffic. VBR traffic usually comes in bursts. If there is no priority between different types of traffic, then long delays could occur while transmitting voice traffic. This can result in poor service quality. Ensuring the existence of priority in the reservation phase of demand assignment MAC protocol can help requests of multimedia traffic to reach the BS with shorter delays.

Among the strategies described earlier, the Request Slot Class Assignment (RSCA) is an effective method to provide a prioritised access scheme to the reservation channel of demand assignment MAC protocols. With the same number of MTs in each class, a class assigned with more request slots will experience shorter delays than a class assigned with fewer request slots. Once the request slots are assigned, they can only be accessed by a given traffic class and are not shared by any other classes. Since only a certain number of request slots are available in the reservation phase of each frame, not all classes can have a large number of request slots. In order to provide lower delays for high priority traffic, lower priority classes are sacrificed. A low priority class can have very few request slots assigned to them.

This can cause long delays before an access to communication channel in granted and inability of the class to support a larger number of terminals.

When request slots are grouped and assigned to different classes, if one class is temporarily under heavy usage, it does not mean that other classes are. While the traffic flow of one class is heavy, the other classes might not have much traffic at all. The request slots that are assigned to these less active classes are unused, so the corresponding channel bandwidth is wasted. If a burst of traffic occurs in a class that has had a very few request slots, the system can become unstable and cause very long delays. To resolve the above problem, we introduce a new scheme for selecting request slots called transmission probability assignment.

4.2.1 Transmission Probability Assignment

Let the transmission probability be the probability of transmitting a request packet in a given request slot at a given time. The packet picks a request slot for transmitting its request within it. In RSCA, the request slots are grouped and assigned to different traffic classes. Only terminals specified from a given traffic class can access pre-assigned request slots. However, in our new assignment scheme, no request slot is assigned exclusively to a particular class of MTs. Slots pre-assigned to a given class can still be used by other classes, however with a specified transmission probability only. Each MT has a transmission probability of 1.0 that is broken apart and assigned to different request slots every time it has a request to transmit. The transmission probability is distributed by an MT according to the usage of each request slot in the past frames. Based on this information, an MT assigns different transmission probability to different requests. A request slot can be accessed by MTs belonging to different traffic classes.

In the RSCA scheme, the request slot assignment can be viewed as the one with transmission probability being assigned evenly to the request slots of a given class. For example in Figure 4.1a, it is assumed that request slots 2 and 3 are assigned, say, to Class 2 traffic, i.e. every MT of Class 2 has a transmission probability of 0.5 for request slot 2 and 0.5 for request slot 3. MTs of Class 2 can only transmit in slots 2 and 3. In the proposed transmission probability assignment scheme, an MT of Class 2 has a chance of transmitting its request packet in other request slots as well (see

Figure 4.1b). It has a higher chance of picking up request slots 2 and 3 because the transmission probabilities for these slots are higher than in others. The main purpose of this assignment scheme is to relieve the load of one traffic class by using the possibly unused request slots that belong to the other classes. However, the delay of the other classes must not rise too much and destroy class prioritisation. Such a situation can be controlled by limiting the amount of transmission probability assigned to these classes.

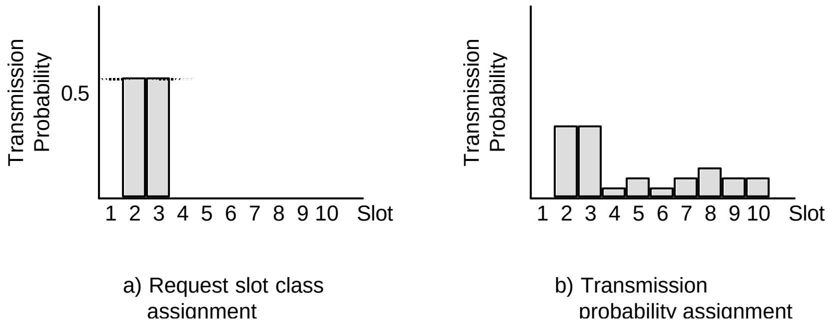

Figure 4.1: Transmission probability in request slots.

Since this scheme is based on the concept of probabilities, it is possible that an MT picks a request slot with heavy traffic to transmit its request. Picking a request slot with heavy traffic load does not mean that a collision will occur there for sure. To avoid the occurrence of selecting a heavy traffic request slot, larger transmission probabilities are assigned to request slots with lower traffic. Random selection based on transmission probabilities is essential for providing a stable system. If MTs always select the request slots of higher transmission probability, then the likelihood of collisions would be much higher than following random selection within all transmission probabilities.

4.2.2 Transmission Probability Based Dynamic Slot Assignment Protocol

The transmission probability based dynamic slot assignment (TRAPDYS) is an access protocol for the reservation phase of a demand assignment MAC protocol. It is improved from RSCA. The TRAPDYS protocol implements the idea of the transmission probability assignment. It assigns transmission probability for an MT dynamically, depending on the current traffic conditions. It is designed to use in the reservation phase of demand assignment MAC protocols. Figure 4.2 shows the frame structure suitable for implementing TRAPDYS. The uplink frame consists of an uplink data transmission phase with multiple data slots and a reservation phase with multiple request slots. The downlink frame contains a header phase and a downlink data transmission phase. The header phase is made up of the synchronization bits, frame format bits, frame assignment bits, and acknowledgement bits for the previous reservation phase. The duplex mode can be either of TDD or of FDD type.

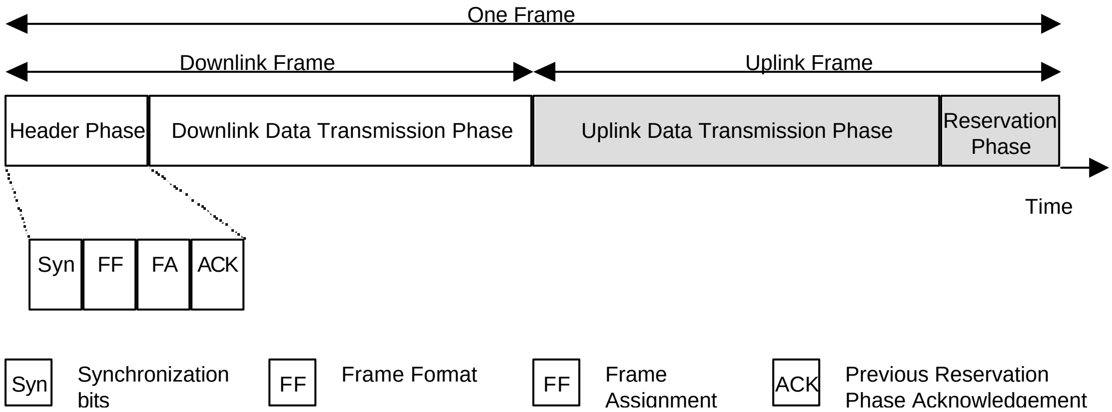

Figure 4.2: A demand assignment MAC frame.

The TRAPDYS protocol observes and uses the traffic information obtained from the past frames to generate a two-dimensional Request Slot Usage table (RSU-table). The table is used to record the status of each request slot over the last N_f frames. Based on this table, the protocol can assign higher transmission probabilities to the request slots with smaller probabilities of collision and increase the chance of transmitting in the less used request slots. The protocol uses past traffic conditions to predict the best location for the next transmission. Figure 4.3 shows the flowchart of the TRAPDYS protocol. An MT observes the traffic activities in the reservation phase by listening to the acknowledgement part of the header phase in the downlink frame. The header phase contains the acknowledgments of the request slots in the previous frame. Each MT maintains its own RSU-table. The table records the activities (a collision, a successful transmission, or no activity) within each request slot in the reservation phase in the past N_f frames.

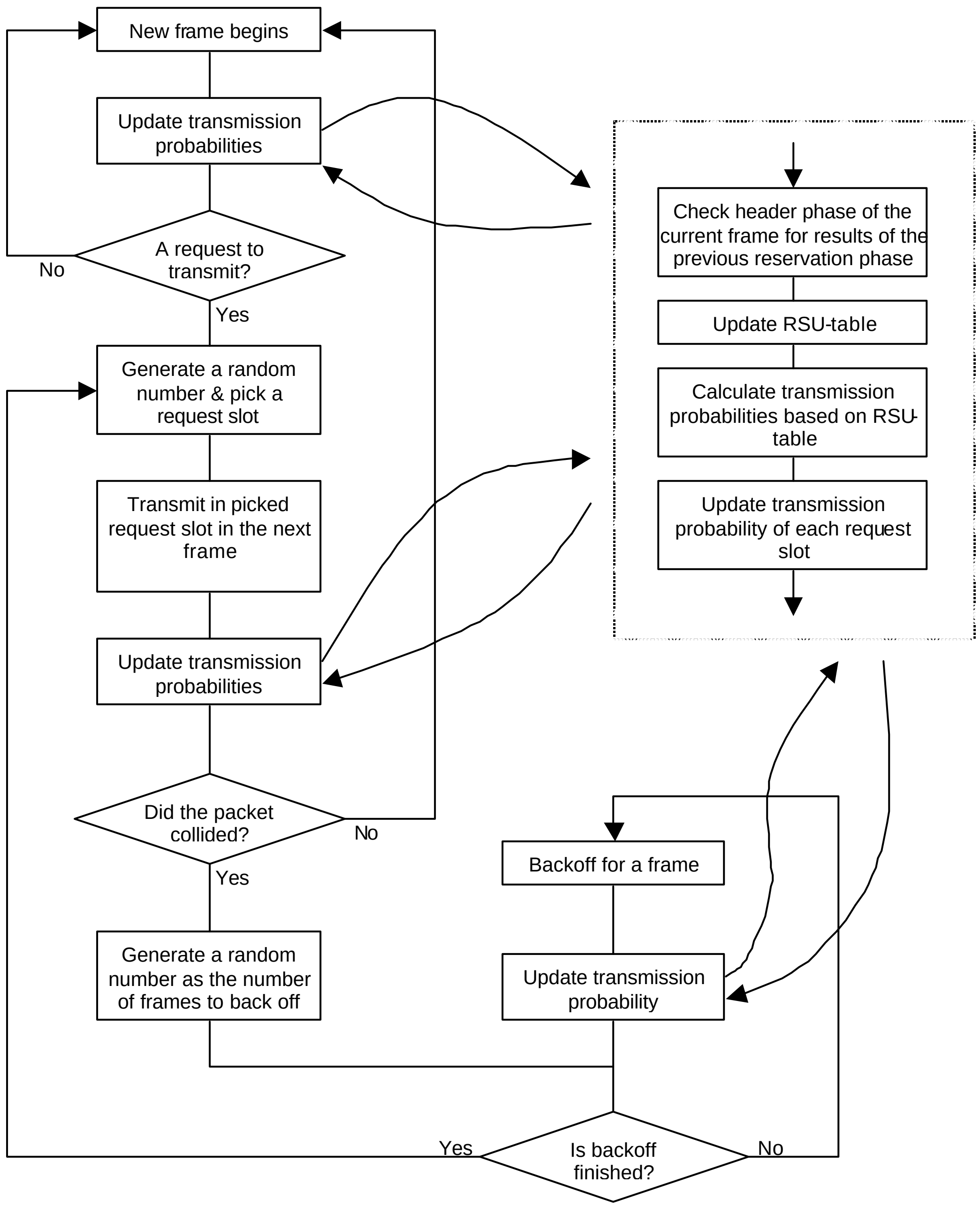

Figure 4.3: TRAPDYS flow diagram.

When a new frame begins, the MT removes the oldest record kept in the last row of the RSU-table and adds a new record of the newly observed activities of the request slots. The actual implementation is quite simple with the help of pointers moving through the table. Note that only a small amount of computation is required from an

MT. Once the RSU-table is updated, the transmission probabilities are re-calculated. The one-dimensional Transmission Probability table (TP-table) contains the transmission probability for each request slot. The number of entries in the table equals to the number request slots in the reservation phase. The sum of transmission probabilities in this table is equal to 1.0. We will discuss calculations required for updating the transmission probabilities later. All the observations, calculations, and assignments are done by each MT individually. The above transmission probability calculation and update are repeated in each frame.

When the MT has a request packet to transmit, it generates a random number P, $0<P=1$. Since the sum of transmission probabilities in TP-table equals to 1, the random number points at one of the request slots. Figure 4.4 is an example of slot selection. The figure shows the TP-table of an MT. There are five request slots, each slot associated with a transmission probability. These probabilities are used to divide the interval (0,1) into subintervals of widths equal to their transmission probabilities, see Figure 4.4. The MT picks the request slot associated with the interval containing a given random number, and uses that for transmitting the request packet in the incoming uplink frame.

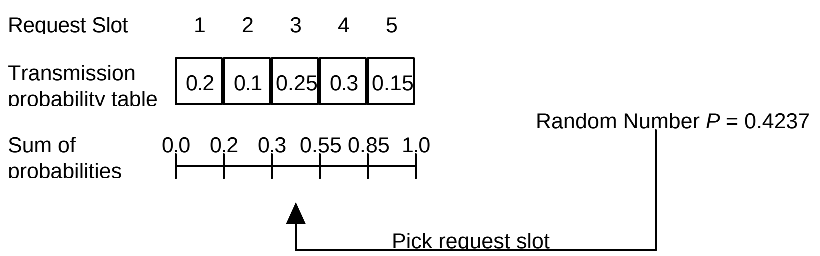

Figure 4.4: Example of transmission slot selection.

Having transmitted its request packet, the MT waits and listens to the next downlink frame for an acknowledgement. If the packet has been received by the BS successfully, then the MT will know it from the header phase of the downlink frame and can transmit its data in the assigned data slot in the uplink data transmission phase. We do not include these steps here in the flow chart but focus on the TRAPDYS protocol instead. The MT goes back to the top of the flow chart and repeats the cycle.

If a collision has occurred while transmitting a request packet, the MT goes into a collision resolution period. The collision resolution in the TRAPDYS protocol is the same as in the slotted-Aloha access protocol. The MT generates a random number between 0 and *B* (*B* is the maximum number of backoff frames). This number indicates the number of fames the MT should back off before retransmitting the collided request packet. The MT then decreases the probability of transmission in request slots that belong to its class. The details will be discussed later with an example. The next three steps are exactly the same as the first three steps in the flow chart depicted in Figure 4.3. The MT observes the channel and updates the TP-table. It repeats these steps until the backoff time is over. Then it generates a random number and retransmits the packet following the cycle discussed earlier. This collision resolution algorithm could be replaced by another collision resolution algorithm, such as the tree collision resolution algorithm discussed in Chapter 3.

4.2.3 Dynamic Adjustment of Transmission Probability

The transmission probability assignment in TRAPDYS has two stages. Each stage produces a transmission probability (P_1 and P_2) for each request slot in the reservation phase. The final transmission probability of each request slot equals P_1+P_2. In Stage 1, the transmission probability P_1 is determined on the basis of current traffic condition in the reservation phase. Stage 2 is responsible for providing a prioritised access scheme. Here, the transmission probability P_2 is determined depending on a pre-defined priority scheme, using the class assignments as defined in the RSCA scheme. Figure 4.5 shows an example where request slot 1 is assigned to Class 1, request slots 2 and 3 are assigned to Class 2, request slots 4, 5, and 6 are assigned to Class 3, and request slots 7, 8, 9, and 10 are assigned to Class 4.

The complete probability of transmission is split into two fractions: an other-class fraction (F_{oc}) and a self-class fraction (F_{sc}). The two fractions are pre-defined before the protocol begins to function. They can be changed during calculations of transmission probability, but are always reset back to the pre-defined value at the beginning of a frame. The sum of the two fractions equals to 1.0.

$$1.0 = F_{oc} + F_{sc}$$

The **other-class fraction (F_{oc})** is the transmission probability assigned according to the current traffic conditions. When F_{oc} increases, relative differentiation between priority classes diminishes. This probability is assigned during Stage 1 of the transmission probability assignment and it gives no preference to any traffic class.

The **self-class fraction *(F_{sc})***, determined during Stage 2, is the component of the transmission probability assigned to the request slots that belongs to the traffic class of a given MT. This is to ensure that different levels of priority are given to different traffic classes. If F_{sc} equals to 0.8, it means that 80 percent of all traffic generated in that traffic class will be transmitted in request slots assigned to that class (we refer to this class as self-class and these slots as self-class request slots). The other 20 percent (F_{oc} = 0.2) can be distributed over request slots originally assigned to other classes (we refer to these classes as other-classes and these slots as other-class request slots) or to the self-class, depending on the current traffic condition. If F_{sc} equals to 1.0, then the performance of the protocol will be exactly the same as the RSCA scheme.

As mentioned, during Stage 1 of transmission probability assignment the value of F_{oc} is established according to the current traffic conditions. According to our proposal, this is done by considering the following factors:

Empty-slots ratio (R_e): This is the ratio of how many empty request slots occurred in the last N_f frames:

$$R_e = \frac{N_e}{N_f}$$

where N_e is the number of times a given request slot was empty during N_f frames. It is calculated for each request slots column by column (slot by slot) from the RSU-table. The empty-slots ratio is a simple indication of the traffic conditions during a given request slot in the past N_f frames. If it is low, then there is a strong possibility of a collision if an MT transmits a packet in that request slot.

Gate fraction (F_g): This is a threshold used to select the request slots with a low traffic flow. The gate fraction functions as an indicator for the MT to decide whether a non-zero transmission probability should be assigned to a given request slot. Any slot with R_e lower than F_g will not be associated with any transmission probability of F_{oc}.

Total number of empty slots (N_{te}): This is the sum of N_e empty slots with $R_e \geq F_g$, observed over last N_f frames.

By looking at the R_e obtained from the RSU-table, the MT can know which slots have carried heavy traffic and which slots have not. The MTs choose the request slots that have low traffic flow, i.e. request slots with $R_e \geq F_g$, and assign a transmission probability according to the formula below. If the request slot has $R_e < F_g$, then a small portion of F_{oc} is taken away and given to F_{sc} in the Stage 2. The size of this portion can equal to F_{oc} divided by the total number of request slots in a frame. This decreases F_{oc} when the number of request slots with $R_e \geq F_g$ is small, and allows more traffic to go through the self-class request slots. If this is not done, the transmission probability can be concentrated within a smaller number of request slots. However, this can cause traffic congestion in these request slots.

Stage 1 transmission probability (P_1) of each request slot is calculated as follows:

$$P_1 = \begin{cases} 0.0\,, \text{ if } R_e < F_g\,, \\ \dfrac{N_e}{N_{te}} F_{oc}\,, \text{ if } R_e \geq F_g\,. \end{cases}$$

No transmission probability is assigned to the request slots with heavy traffic flow where $R_e < F_g$. If the request slot has $R_e \geq F_g$, then its transmission probabilities are updated. Figure 4.5 shows an example on how transmission probabilities are assigned in Stage 1. In the example, the protocol's memory spanned over 10 frames ($N_f = 10$), so the RSU-table records the number of empty slots in the last ten frames. R_e is calculated by dividing the number of empty slots in request slots over the last ten frames by 10. The stage 1 of transmission probability assignment then begins. Any request slot that has R_e lower than F_g ($F_g = 0.5$) is assigned a zero transmission probability in our example, there are three request slots with $R_e < F_g$ (request slots 1, 2 and 3). The original F_{oc} equals 0.4. After subtracting the portion of the three request slots (3×0.4÷10=0.12), F_{oc} equals to 0.28 (0.4-0.12=0.28). Using the total number of empty slots with R_e<F_g, we have N_{te} = 6+5+6+6+8+7+9 = 47, and the Stage 1 transmission probability of slot 4 becomes $P_1 = 6 \div 47 \times 0.2.8 = 0.036$

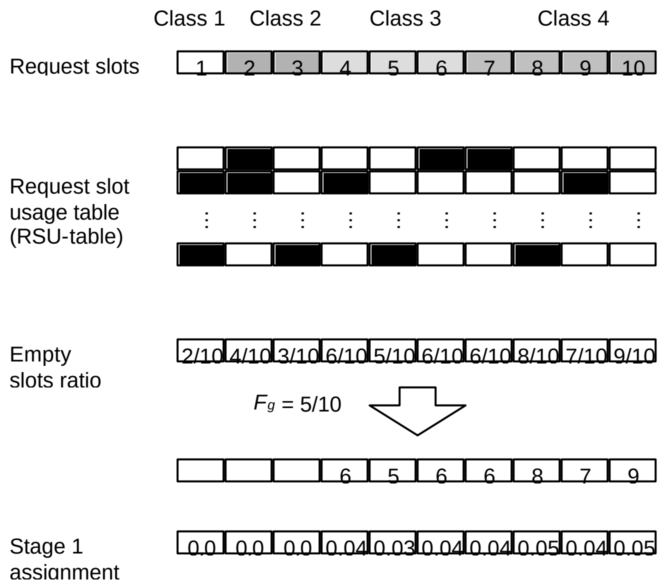

Figure 4.5: Example of Stage 1 of transmission probability assignment.

In the Stage 2 of transmission probability assignment, F_{sc} is divided equally and assigned to the request slots that have been assigned the traffic class of a given MT. For example, if the MT belongs to class 3, then the request slots of its self-class are slots 4, 5, and 6 (Figure 4.6). F_{sc} is assigned to these three request slots evenly. Before F_{sc} is spread over these slots, it is increased by the transmission probability from request slots that had $R_e < F_g$ in Stage 1. Then Stage 2 transmission probability (P_2) is assigned as follow:

$$P_2 = \begin{cases} 0.0, \text{ if the request slot belongs to other class ,} \\ \dfrac{F_{sc}}{N_{sc}}, \text{ if the request slot belongs to self class .} \end{cases}$$

where N_{sc} is the number of request slots belonging to the class of a given MT. An example in Figure 4.6 shows the Stage 2 of transmission probability assignment. The request slots have been grouped into four classes, each with a different number of request slots. In the case of class 3, F_{sc} (= 0.72) is divided by 3 and is equally spread over request slots 4, 5, and 6 (0.24 each). Other request slots are associated with the transmission probability of zero.

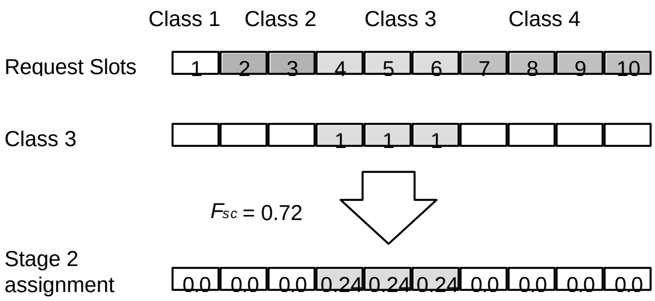

Figure 4.6: Example of Stage 2 of transmission probability assignment.

When the two stages of computations are completed, the transmission probabilities produced for each request slot are combined. Figure 4.7 shows the final transmission probability of each request slot.

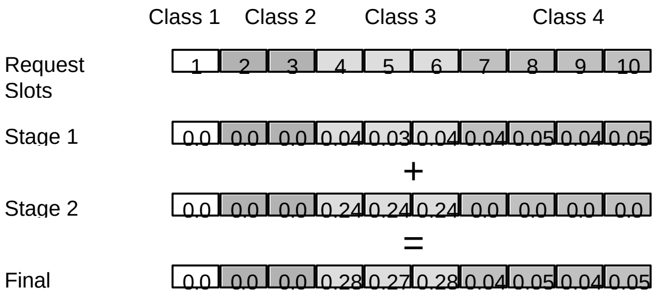

Figure 4.7: An example of transmission probability assignment.

As we see in this example, the MT has a chance to transmit in slot 4 to slot 10. Most transmission probabilities are distributed in slots 4, 5, and 6. This is not surprising since this MT belongs to Class 3. Slots 1, 2, and 3 are under heavy traffic flow and are not assigned any transmission probability.

4.2.4 Key Design Issues of TRAPDYS

In this section, we discuss some of the problems encountered while formulating the TRAPDYS protocol.

Maintaining the priority scheme

The TRAPDYS protocol is an extension of RSCA. Its goal is to utilize request slots that are possibly free, but at the same time to maintain an assumed prioritised access scheme. In RSCA, the access priorities are defined by assigning different numbers of request slots to different classes. TRAPDYS maintains a given priority access scheme by assigning F_{sc} to appropriate request slots. If F_{sc} is large, the chance of transmission in the request slots that are assigned to the self-class is high and the prioritised scheme can be maintained.

Validity of traffic observation

The protocol assigns a transmission probability based on the traffic observed during the last N_f frames. This method is very simple and direct. The assumption is that if a request slot has been used heavily in the past frames then the chance of it being occupied in the next frame is high. One could argue that since the protocol is based on random access, the measurement done during past frames can be uncorrelated with traffic observed on the current frame. This is true in some aspects. If an MT has just sent a request, it is not likely to transmit again in the near future, unless its request packet has collided. From this point of view, there are no correlations between occupations of request slots on consecutive frames. However, the observation focuses on the traffic density rather than individual transmissions, and the traffic density is related to the number of active MTs. The traffic density increases when the number of active MTs increases. Some occasional bursts can also increase the traffic density for a short period.

The traffic density can be calculated by observing empty slots over the last N_f frames. If the value of N_f is small, then reaction to changes occurring in traffic density can be quicker. On the other hand, large N_f could mean that less relevant (outdated) information is still taken into account when calculating transmission probabilities. Selection of the most appropriate value of N_f can be a complicated issues and an engineering compromise may be needed in practical implementation. A measurement based on a threshold is proposed to allow easy implementation and to decrease resources consumption. Complex prediction schemes are unlikely to be useful in short periods of time.

Feasibility of traffic observations

The TRAPDYS protocol requires MTs to maintain records of transmissions over last N_f frames. Thus, each MT has to monitor the state of consecutive frames. In a well-designed system, this can be done without wasting much resource. An MT only has to switch on and listen to header phases at the beginning of downlink frames. The beginning of the header phase usually contains some synchronization bits, frame format information, and frame assignment information. The synchronization bits allow MTs to synchronize with BS, and the frame format information gives the MTs ideas about the structure of the entire frame. The frame assignment information tells MTs if there are data packets for them from the BS and if they have been given the right to use uplink data slots that they have requested. MTs usually power down after such an overhead phase if there is nothing transmitted for them, until the next downlink frame. If the acknowledgements for the request slots of previous frames are placed directly after these overheads at the end of the header phase (see Figure 4.2), MTs are only required to switch on for a little longer to obtain the acknowledgement information. MTs are not required to be switched on at all times. Therefore, observing the channel does not consume much power.

4.3 Summary

In this chapter, we have discussed the strategies that have been used by various demand assignment MAC protocols to further improve their efficiency. The simplest method for providing random access with priorities is to assign request slots to different classes of traffic. Building upon this method, we have introduced the concept of transmission probability. This concept allows a request slot to be assigned to many different traffic classes at the same time. Next, we have proposed the TRansmission Probability based DYnamic Slot assignment (TRAPDYS). The TRAPDYS protocol operates dynamically by observing the traffic conditions. It uses information about the recent traffic conditions to assign a transmission probability with which an MT can select request slots with lower traffic. The protocol is executed by each MT. Implementation of the protocol does not consume any resources of the BS. Each MT functions independently with its own transmission probability assignment.

Chapter 5

Performance Evaluation

In this chapter, we evaluate the performance of our transmission probability based dynamic slot assignment (TRAPDYS) protocol using quantitative stochastic simulation. We focus on the performance of the TRAPDYS algorithm against the performance of the Request Slot Class Assignment (RSCA) scheme discussed in Chapter 3, and introduced in [Jain99, Jian98, Haba00, Zhij00]. There are several issues that need to be observed in a credible quantitative simulation study [Pawl02]: use of a reliable pseudo-number random generator, selection of the right type of simulation, use of a correct method for analysing the output data, and obtaining a low statistical error of the final results. In order to produce credible simulations with such elements, we used a simulation package called AKAROA-2 [Ewin99]. It is controller of a quantitative stochastic simulation able to terminate a simulation automatically when errors in the simulation results are lower than a desired level.

5.1 Simulation Model and Assumptions

We have created a simple model that allows us to directly compare TRAPDYS and RSCA. A more complex and realistic model could be considered when one wishes to analyse performance of the TRAPDYS protocol in a particular application. Since TRAPDYS is a modification of RSCA, the same assumptions are used when simulating both protocols. The simulation model and assumptions can be summarised as follow:

A1. The analysed network is centralized with one BS in the middle and many MTs surrounding the BS.

A2. A TDD frame structure of the two protocols is defined in Figure 5.1. Its format is similar to that of MASCARA [Mikk98]. The frame consists of a header phase, a downlink data transmission phase, a reservation phase, and an uplink data transmission phase. It is assumed that this frame is transmitted during 5 ms. The header phase of the frame contains the bits used for synchronization, the frame format information, frame assignment, and the acknowledgement of the reservation phase of the last uplink frame. The reservation phase consists of ten mini-slots for most simulation studies, chosen for simulation convenience.

The ten mini-slots are divided into four priority classes. Request slot 1 is for class 1. Class 4 has the highest priority and Class 1 has the lowest priority. Request slots 2 and 3 for Class 2. Request slots 4, 5, and 6 for Class 3. Request slots 7, 8, 9, and 10 for Class 4. A study of the effects of different numbers of classes and request slots in the reservation phase uses a slightly different layout (see Study 5).

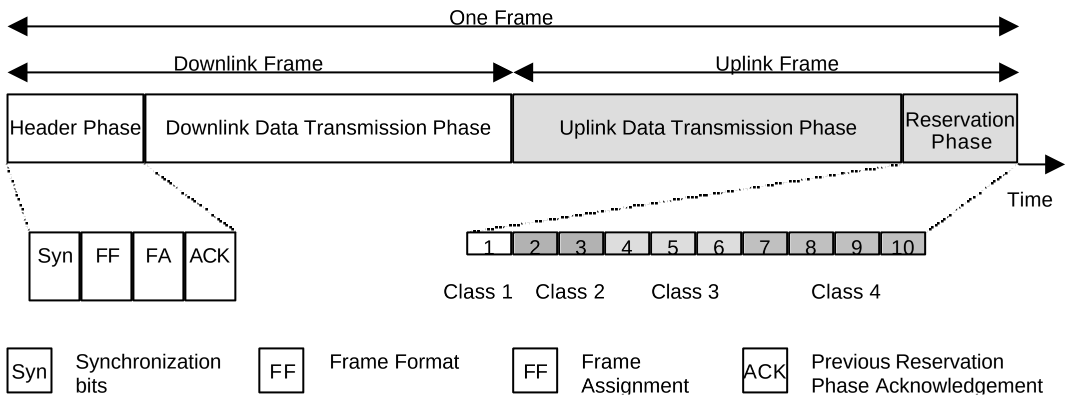

Figure 5.1: Frame structure used in TRAPDYS algorithm simulation.

A frame length of 5 ms is chosen with no specific reason. A frame length smaller than 20 ms is common in large wireless networks [Pras98]. If a communication channel has a bandwidth of 20 Mbits/s, then the TDD frame is 102.4 Kbits in length. MASCARA frame formation is used in our simulation. MASCARA is a demand assignment protocol with a clean structure. We use the frame structure of MASCARA to show that TRAPDYS can be easily implemented in the existing demand assignment protocols. In the reservation phase, the request slots are assigned to different classes to produce a stepwise priority scenario. This helps us to observe the behaviours of TRAPDYS and RSCA.

A3. The slotted-Aloha access protocol is used both as the random access scheme of TRAPDYS and RSCA. The collided MTs apply a collision resolution algorithm in which they schedule the next retransmission by choosing K frames to back off randomly from between 0 and $K_{max} = 10$.

The assumed backoff algorithm is the simplest collision resolution algorithm applied in slotted-Aloha protocol. Influence of K_{max} on the performance of slotted-Aloha is discussed in [Hamm86].

A4. The BS has no problem with assigning bandwidth to the MTs that submitted the requests successfully.

A5. We assume a perfect environment where no interference exists. Each MT is equipped with a transceiver that has a perfect power control. Capture effects, transmission errors, and propagation delays do not exist. Synchronization, scheduling, downlink broadcast, and downlink data transmission are assumed to work perfectly and are not simulated in detail here.

For example, once a request is accepted by a BS, it is up to the BS to schedule a data transmission. By implementing such actions into our simulations, one would introduce undesirable extra variables. We are interested only on the performance of TRAPDYS and RSCA.

A6. MTs belong to different priority classes; and each MT generates one type of traffic only.

A7. All MTs generate real time voice VBR traffic only. The traffic of real-time VBR is modelled by Markov Modulated Poisson Process (MMPP) with two states: ON and OFF. ON time = 1 sec (the mean time spent in State ON), OFF time = 1.3 sec (the mean time spent in State OFF), and data rate in ON state = 8 kbps [Fisc92].

The assumption that each MT generates one type of traffic only makes performance evaluation studies of TRAPDYS and RSCA easier to compare. This allows us to observe the relative performance of each class under different protocols. The traffic density can simply be changed by changing the number of MTs. VBR traffic is chosen because it has high demand and requires extra care to satisfy its variable bit rate nature.

A8. Steady-state simulations are used in Study 1, Study 2, Study 3, and Study 5 (see next section). They are the results of simulation models running for a long amount of time (in theory, an infinite amount of time). Terminating simulations are used in Study 4 with specified lengths of time.

We will present numerical results obtained from steady-state simulation to assess behaviour of the TRAPDYS protocol and the RSCA protocol over a long time of operation. Steady-state simulations are done using the AKAROA-2 simulation package and the results were collected at a 95 percent confidence level with a relative statistical error not greater than 0.01. Terminating simulations are used to observe the behaviour of the two protocols in a temporarily congested network.

5.2 Simulation Studies and Results

Five simulation studies reported in this section were designed to observe the performance of the TRAPDYS algorithm under different operational conditions. The first study was conducted for finding a satisfactory level of relative statistical error to make our simulation results credible. Next, in the second study, we focus on the dependence of the TRAPDYS performance on its basic variables. In the third study, two cases are analysed to compare the TRAPDYS protocol and the RSCA protocol. Then, we study the dynamic nature of the TRAPDYS protocol and look at the behaviour of the protocol under bursts of traffic. The last study investigates the effects of different numbers of classes and request slots. All results are presented with a 95% confidence level and a statistical error of 0.01 or less, unless otherwise stated.

5.2.1 Study 1: Precision of Results

Any simulations involving random phenomena produce random output data. It is important to gather a satisfactory large sample of such data and analyse them with proper statistical methods. Confidence intervals at a given confidence level are commonly used to assess precision of the final simulation results. We use a simulation package AKAROA-2 to carry out sequential simulation, which is stopped when the sample size reaches the desired level of statistical error. In this study, we set the

TRAPDYS parameters to F_{sc}=0.8 (self-class fraction), N_f =20 (number of past frame over which traffic is observed), and F_g=0.8 (gate fraction). We analyse the average number of frames required for transmission of requests, assuming from 10 MTs to 100 MTs in each class, while changing the relative error of the final results. This study investigates the relationship between the levels of statistical error of results and their usefulness for drawing quantitative conclusions about the analysed system.

Our results are shown in Figure 5.2. The x-axis is the number of MTs (VBR terminals) in the reservation phase in each of the four classes. Each class has the same number of terminals. The y-axis is the average number of frames that are required for an MT to make a reservation. The four curves in the figure represent the same results obtained with different levels of relative statistical error: 0.01, 0.05, 0.10, and 0.50 at 0.95 confidence level. Only simulation results with error of 0.01 or less seem to be acceptable. The other results are very variable, with values much higher than the plot with statistical error of 0.01. The results with higher errors cannot be really used in comparison studies. On the basis of these observations, all steady-state results presented in this thesis have their relative statistical errors not larger than 1%, at 0.95 confidence level.

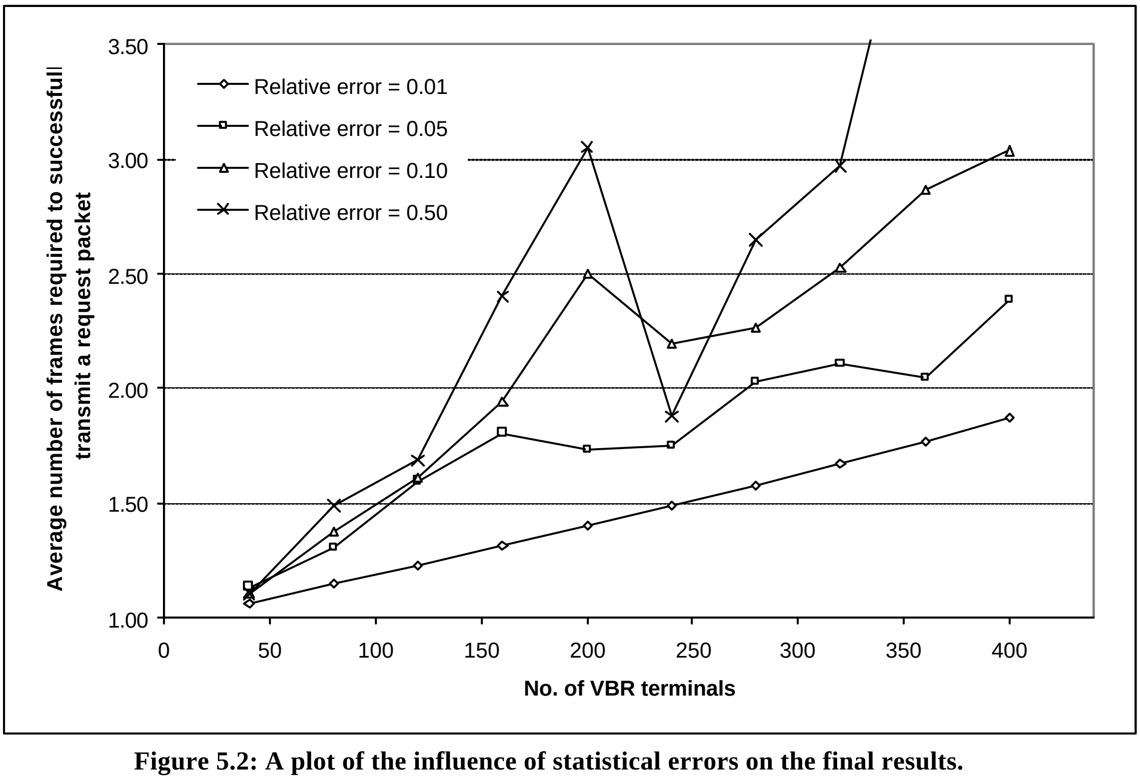

Figure 5.2: A plot of the influence of statistical errors on the final results.

5.2.2 Study 2: Performance of the TRAPDYS Protocol

In this section, we study the effects that different parameters of TRAPDYS have on its performance. There are three important parameters: the self-class fraction (F_{sc}), the gate fraction (F_g), and the size of the observation window (N_f). The influence of each parameter is studied by varying its value while keeping other parameters unchanged. If F_{sc} is high, then the differences in delay of successful request packet transmission are large for each class. A priority scheme is clearly formed. If F_{sc} is low, the delays of all classes merge and the priority scheme disappears. The overall delay should be smaller when F_{sc} is small. F_g is the threshold that the request slots are required to meet before they are assigned any transmission probability. A high F_g means only the request slots with low traffic density are assigned a transmission probability. If a request slot cannot reach the threshold, it is likely to have a high traffic flow in the upcoming frame. The size of the observation window N_f controls the reaction speed of the transmission probability distribution to the current traffic condition. A fast reaction can be obtained with a smaller N_f. If N_f is too small, then the prediction is not accurate.

The results for the F_{sc} simulation are similar to the description above. The priority scheme disappears when F_{sc} is low. Figure 5.3 shows the average number of frames required for a successful transmission when F_{sc} is 0.9. Ninety percent of the transmission probability is assigned to the request slots that belong to the class of the transmitting MT. The figure shows a clear separation between the plot of each class. MTs of Class 4 experience shortest delay waiting for a successful request reservation and MTs of Class 1 experience the longest delay. F_g stays the same throughout the simulation.

When F_{sc} equals to 0.5, fifty percent of the transmission probability of an MT is assigned based on the current traffic conditions. In comparison to Figure 5.3, the gaps between the plots of the four classes in Figure 5.4 are smaller. The differences between Class 3 and Class 4 become blurred. Figure 5.5 shows the result for the four classes when F_{sc} equals to 0.1. Only ten percent of the transmission probability is left for priority scheduling. Most of the transmission probability is assigned according to the traffic conditions. The priority scheme between each class is pretty much diminished. All four classes perform in a similar fashion.

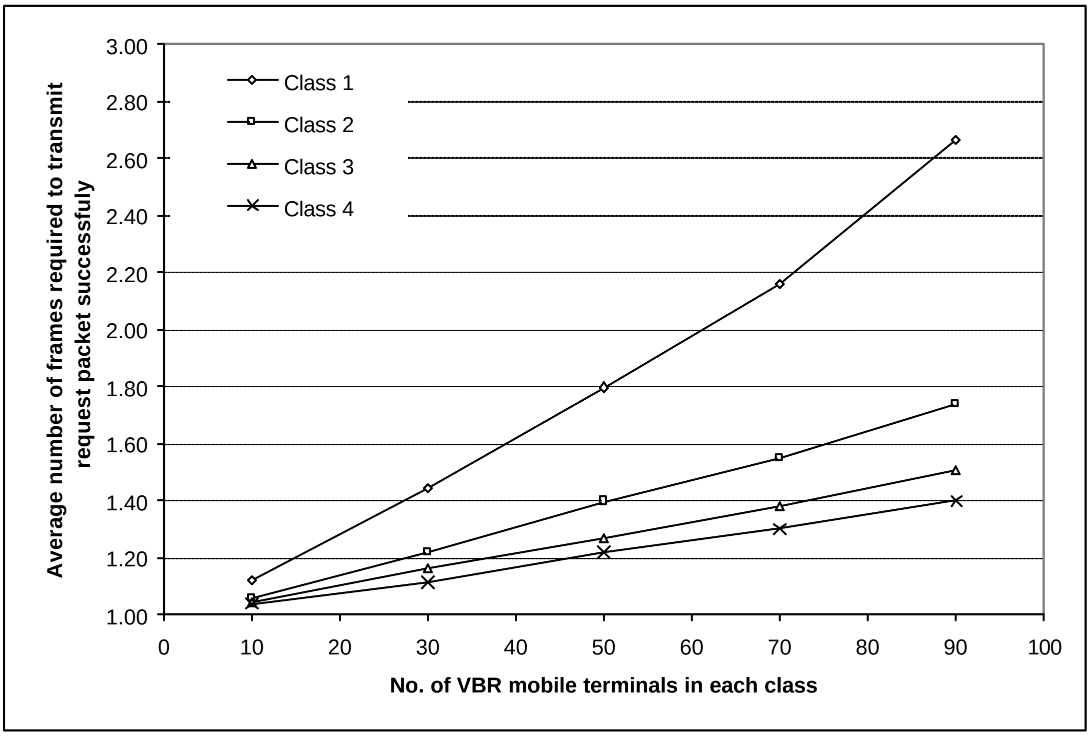

Figure 5.3: A plot of the average delay in the four classes when F_{sc} = 0.9.

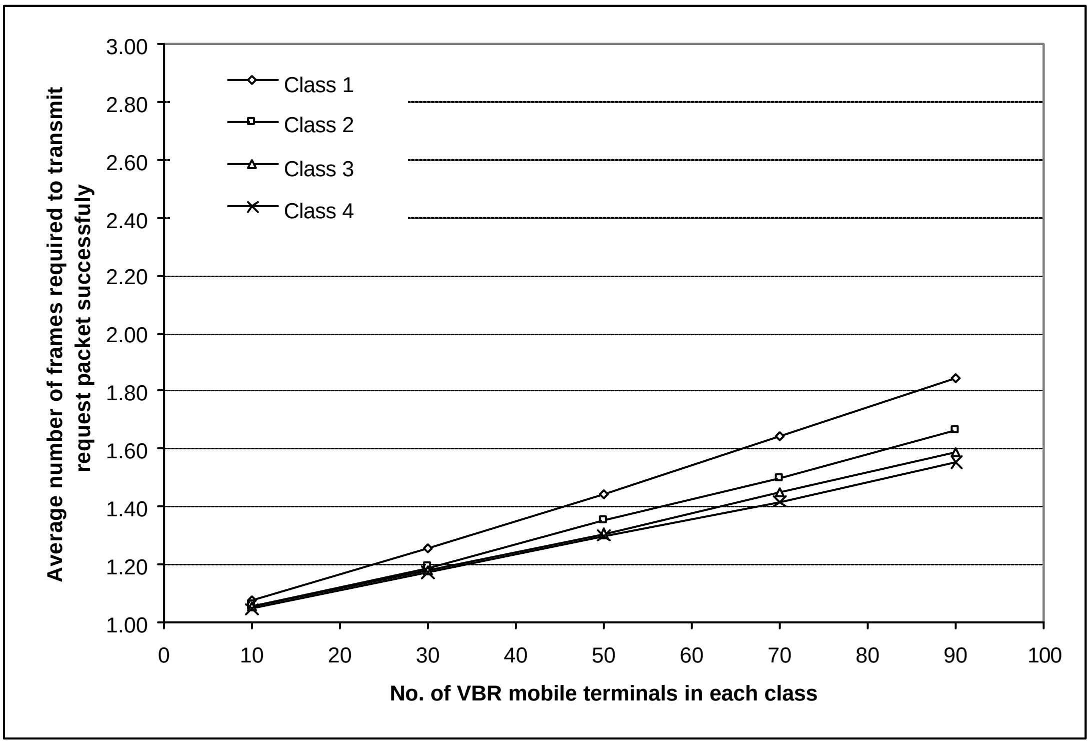

Figure 5.4: A plot of the average delay in the four classes when F_{sc} = 0.5.

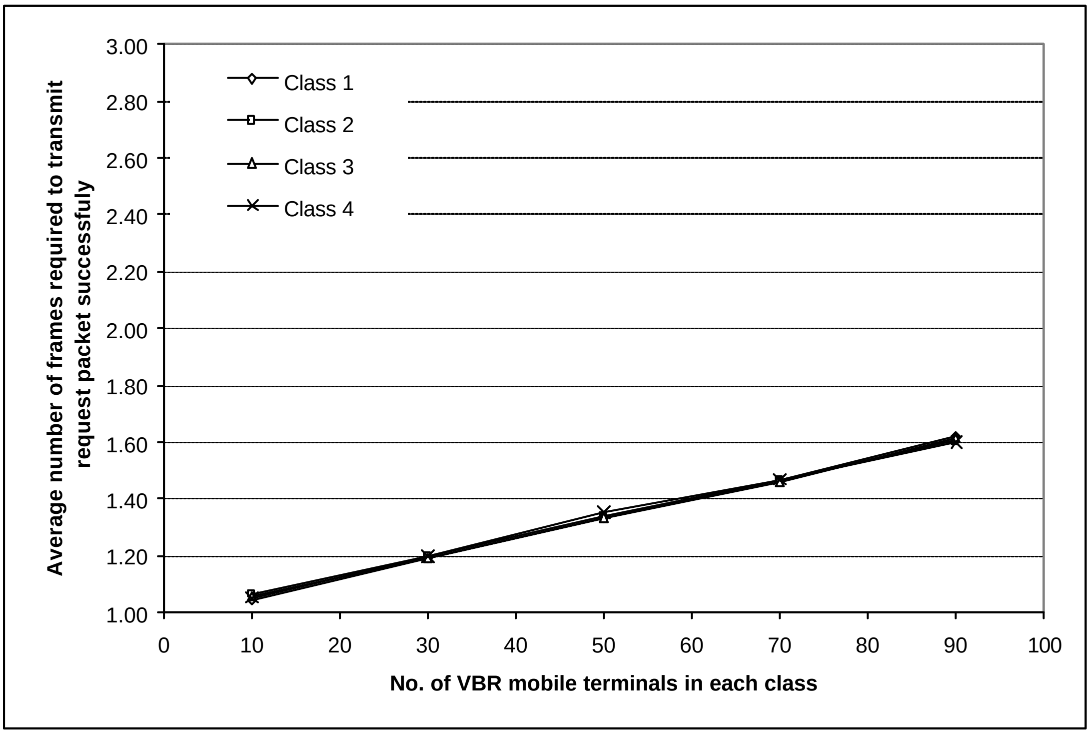

Figure 5.5: A plot of the average delay in the four classes when F_{sc} = 0.1.

F_g is a threshold for deciding whether a request slot should be given a transmission probability in a Stage 1 transmission probability assignment. If the traffic level experienced by a given request slot is high, then a transmission probability should not be assigned to it. Figure 5.6 shows the results obtained when simulating performance of TRAPDYS under different values of F_g.

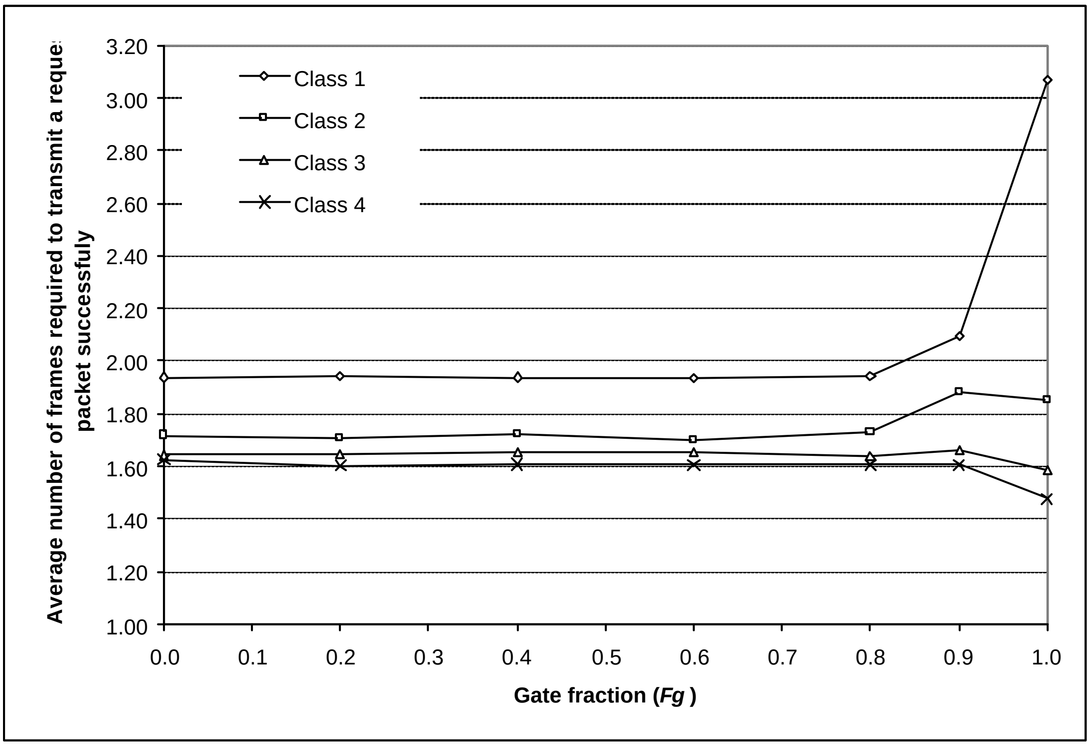

Figure 5.6: A plot of the average delay vs. the gate fraction (F_g).

The TRAPDYS parameters used for this simulation have different values from other simulations. F_{sc} = 0.5 is used to allow a clear observation of the effect F_g has on the network performance, and a large population of MTs is used (100 MTs for each class). The later is assumed to increase the traffic flow. If the traffic flow is low, F_g is not likely to make much difference to the final result. Figure 5.6 shows the quality of service offered by TRAPDYS to each of the four classes when the value of F_g is varied. The mean delays do not show much variation until F_g passes 0.8. The mean delays for Class 1 and Class 2 rise significantly, while the delays for Class 3 and 4 decrease. In this scenario, the traffic flow is high in most classes. If F_g is set high, then most of the transmission probability is assigned to the request slots of the self-class. Little utilization of the request slots of other classes can occur. When F_g equals 1.0, an MT cannot use the request slots of other classes unless there is no transmission over the last N_f frames. The chance of this is small in heavy traffic. Therefore, most of the transmission probability that was supposed to be assigned to other classes is assigned back to the request slots of the self-class. This causes the delay of Class 1 and 2 to rise sharply because a large amount of traffic is concentrated in a small number of request slots. A low F_g value does not cause different classes to merge, as in the case of F_{sc}. The effect of F_g on TRAPDYS performance is more subtle and difficult to observe.

Figure 5.7 shows the results obtained when the number of past frames (N_f) increases from 10 to 400, for F_{sc} =0.5 and F_g =0.9. Again, F_{sc} = 0.5 is used to increase sensitivity of TRAPDYS to N_f. Each class contains 100 MTs. The results for Class 1 and Class 2 show a gradual decrease of the mean delay for successful request transmission when N_f increases. This delay in the Class 1 continues to decrease until N_f reaches 350. In Class 2, the decrease stops when N_f reaches 200. Little variation is observed for Class 3 and Class 4. In this simulation, a large number of observations can help the MTs to assign a transmission probability to the correct request slots. This is unlikely to occur in every case. The traffic density simulated here is very steady without many changes; therefore, a large number of observations is beneficial. If the traffic density fluctuates greatly, then a large number of observations can hinder the transmissions.

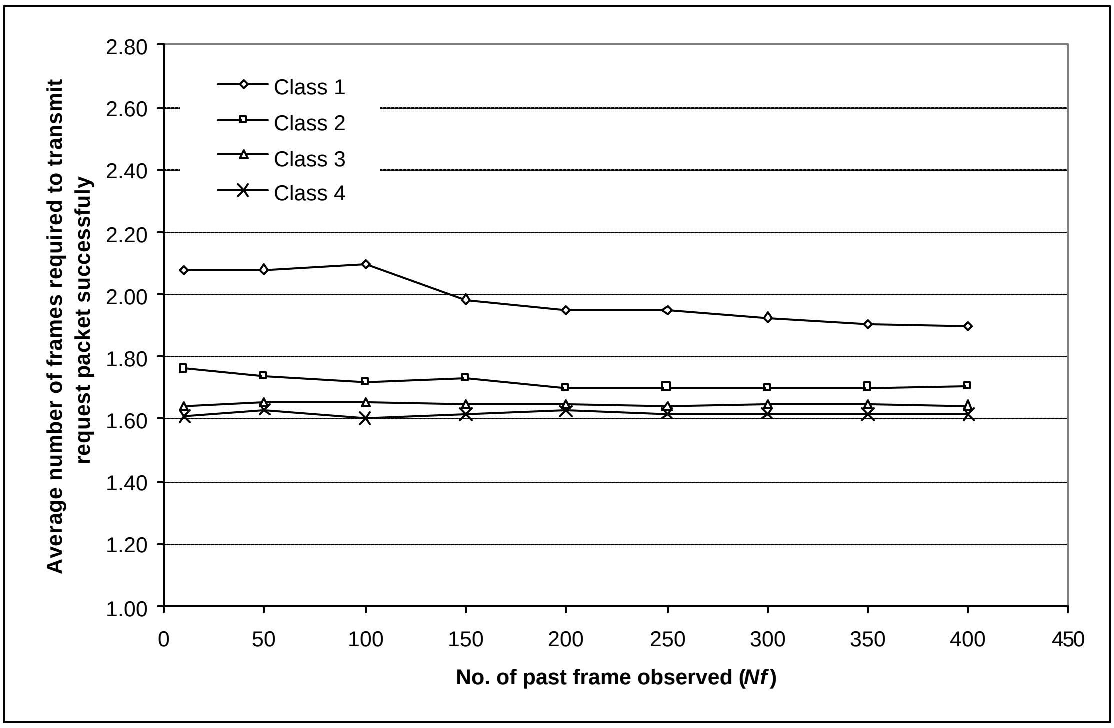

Figure 5.7: A plot of the average delay vs. the number of past frames observed (N_f).

5.2.3 Study 3: Comparing TRAPDYS and RSCA

We consider two cases here. In the first case, the performance of TRAPDYS and RSCA are analysed in a stable environment. The TRAPDYS algorithm is expected to maintain different levels of priority between the classes. The four priority classes have the same number of MTs (VBR terminals). The number of MTs is increased slowly to observe the quality of service experienced by the four classes in the two protocols under different traffic loads. The TRAPDYS parameters are F_{sc} =0.8, N_f =20, and F_g =0.8. In RSCA, we expect Class 4 to have the shortest delay and Class 1 to have the longest delay.

The second case study focuses on the performance of the two protocols when one class is under heavy traffic flow, to observe the ability of the TRAPDYS protocol to distribute traffic load evenly. TRAPDYS is expected to distribute some of the traffic in a heavily loaded class to the classes with less loaded traffic. In the study, Class 3 is given a heavy traffic flow. The number of MTs in the class is set to 100 throughout the simulation. The other classes (Classes 1, 2, and 3) increase the number of their MTs slowly. The TRAPDYS parameters are F_{sc} = 0.8, N_f = 20, and F_g = 0.8.

Case one

Figure 5.8 shows the results for RSCA simulation in the four classes of mobile terminals. The figure shows the average number of frames required to successfully transmit a request packet (the delay of the request packet) as a function of the number of MTs in each class. The delays experienced by each of the four classes are very different. As expected, Class 4 has the lowest delay among the four classes, as it has the same number of MTs as the other classes, but has more request slots assigned. Four request slots are assigned to Class 4 MTs. In comparison to Class 1, the capacity of Class 4 is about four times greater.

These results clearly show a limitation of RSCA. One can see a sharp rise in the mean delay of request packet when the number of MT reaches a certain number N_{max}. We call this point a break-off point. The point before the sharp rise is the absolute capacity of the class. When the traffic density increases to a certain level, the request slots of a given class suddenly become saturated and the collisions cannot be resolved in a short period if there are more terminals than N_{max}. This behaviour can cause the class channel to become unstable. It is desirable not to have this type of break off behaviour happen when the MT population is small. The simplest way to delay the occurrence of such a problem is to limit the number of MTs assigned to the class or assign more request slots to the class. Although both approaches can move the break off point to a later stage, they can waste a large amount of bandwidth if the traffic flow is low. For example, a large traffic burst could cause a break-off to occur. A BS is disrupted and loses all contact with its MTs. After the BS recovers from the disruption, the MTs have to reconnect to the BS. Each reconnection has to be requested through a random access channel. The channel consists of relatively very few request slots. If the BS supports hundreds of MTs, then a break-off will occur in the random access channel that is responsible for connection initiation.

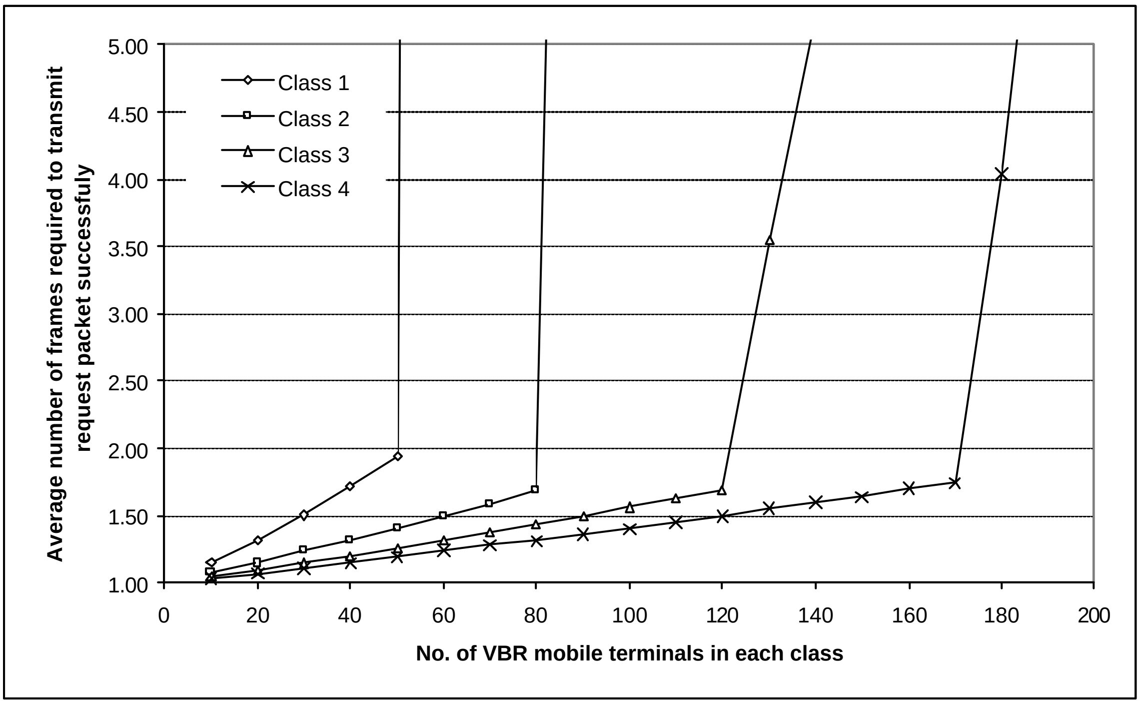

Figure 5.8: Average delay of request in four classes of RSCA.

Under the same circumstances, the performance of the TRAPDYS algorithm is rather different (see Figure 5.9). The priority scheme is still maintained: Class 4 has the shortest delay and Class 1 has longest delay. The differences in delay experienced by different classes are small. The break-off points in TRAPDYS occur much later than in RSCA. Classes 1, 2, and 3 break off at 160 MTs. Class 4 breaks off at 180 MTs.

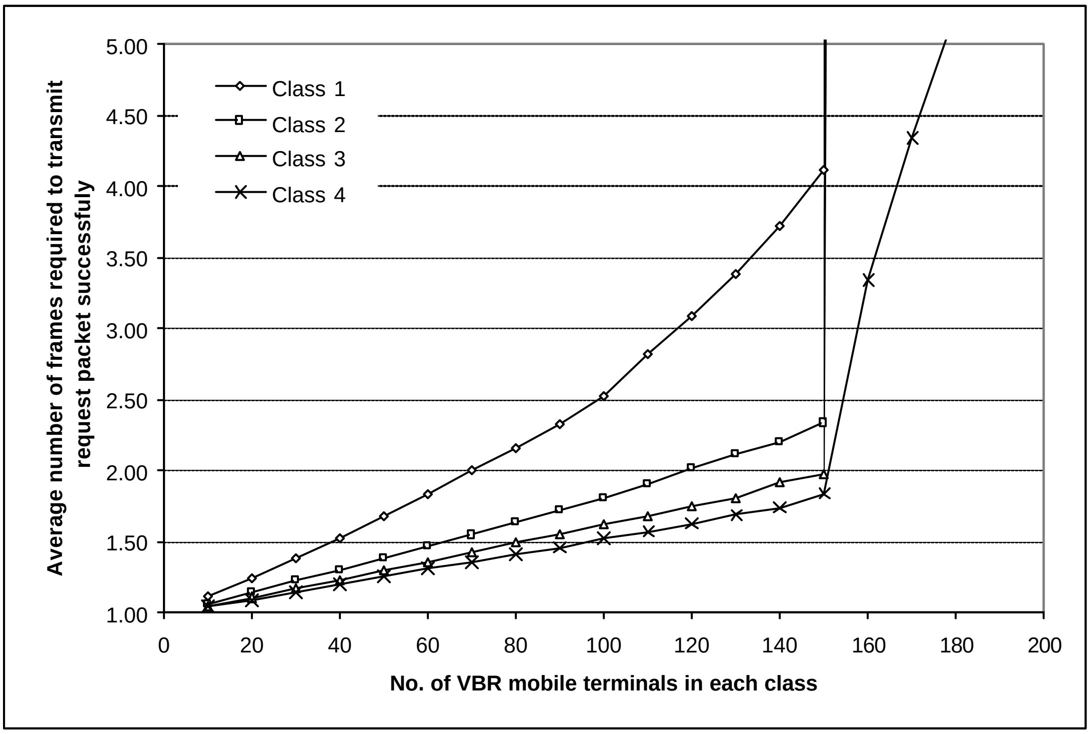

Figure 5.9: Average delay of request in four classes of the TRAPDYS protocol.

Figure 5.10 shows the quality of service experienced by Class 1 and Class 2 under RSCA and TRAPDYS. In Class 1, MTs under TRAPDYS have shorter delays than under RSCA. The size of the gap between the two protocols increases when more MTs exist in the class. TRAPDYS does not have a break-off point at 60 MTs, instead, a smooth line is observed and does not break off until 160 MTs. In class 2, TRAPDYS causes shorter delays than RSCA. The mean request delay under RSCA breaks off for 90 MTs, while under TRAPDYS, it smoothly inclines and breaks off at 160 MTs. Figure 5.11 shows the quality of performance offered for terminals of Class 3 and Class 4. One can see that RSCA actually outperforms TRAPDYS. This is because under TRAPDYS, some of the bandwidth that belong to Class 3 and Class 4 has been used by Class 1 and Class 2. The usage of the Class 3 and Class 4 request slots is increased and hence results in more collisions and longer delays. This behaviour is beneficial if the slight rise of the access delay is acceptable in these two classes. If a traffic class requires very short delays and cannot tolerate the delays produced in TRAPDYS, it should use RSCA. Thus, no other classes can use its request channel. A slight increase of delay in a high priority class can be caused by the decrease of the delay in a low priority class by a large margin.

Figure 5.12 shows the average delay of the four classes for both TRAPDYS and RSCA. The results are averaged from the four classes. A sharp rise occurs at 50 MTs in the case of RSCA. This is due to the break-off point in Class 1 at 50 MTs. The result shows that TRAPDYS performs slightly better than RSCA.

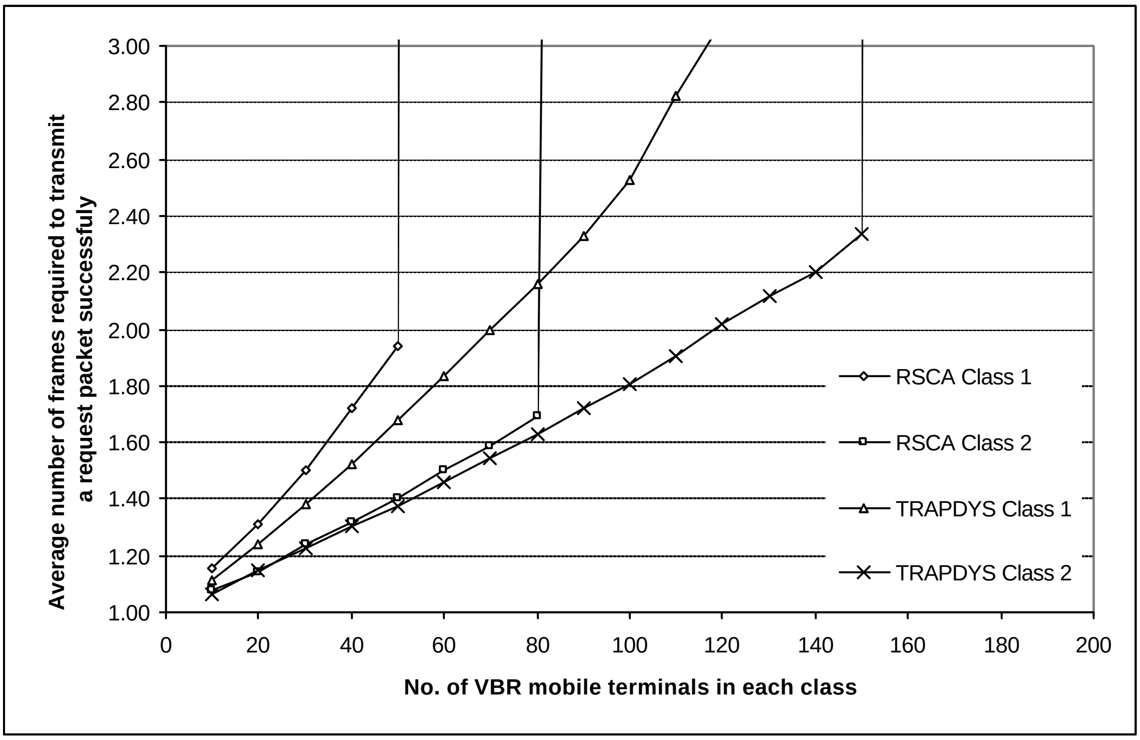

Figure 5.10: Average delay of Class 1 and Class 2 of RSCA and TRAPDYS.

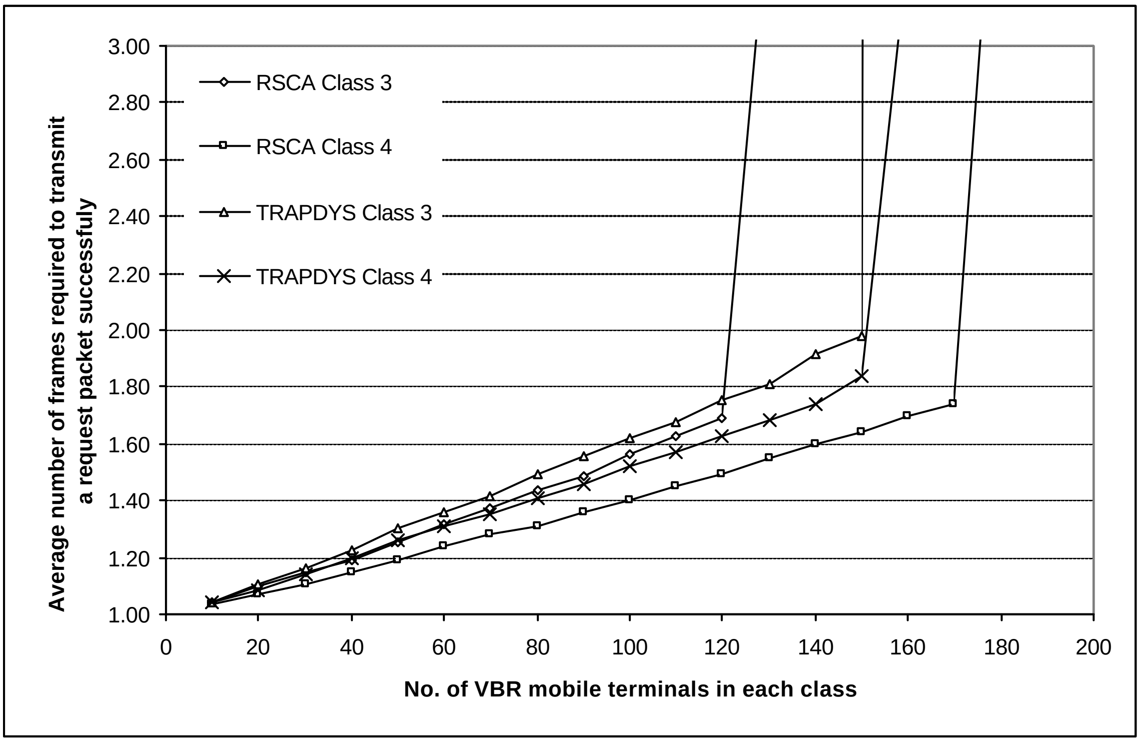

Figure 5.11: Average delay of Class 1 and Class 2 of RSCA and TRAPDYS.

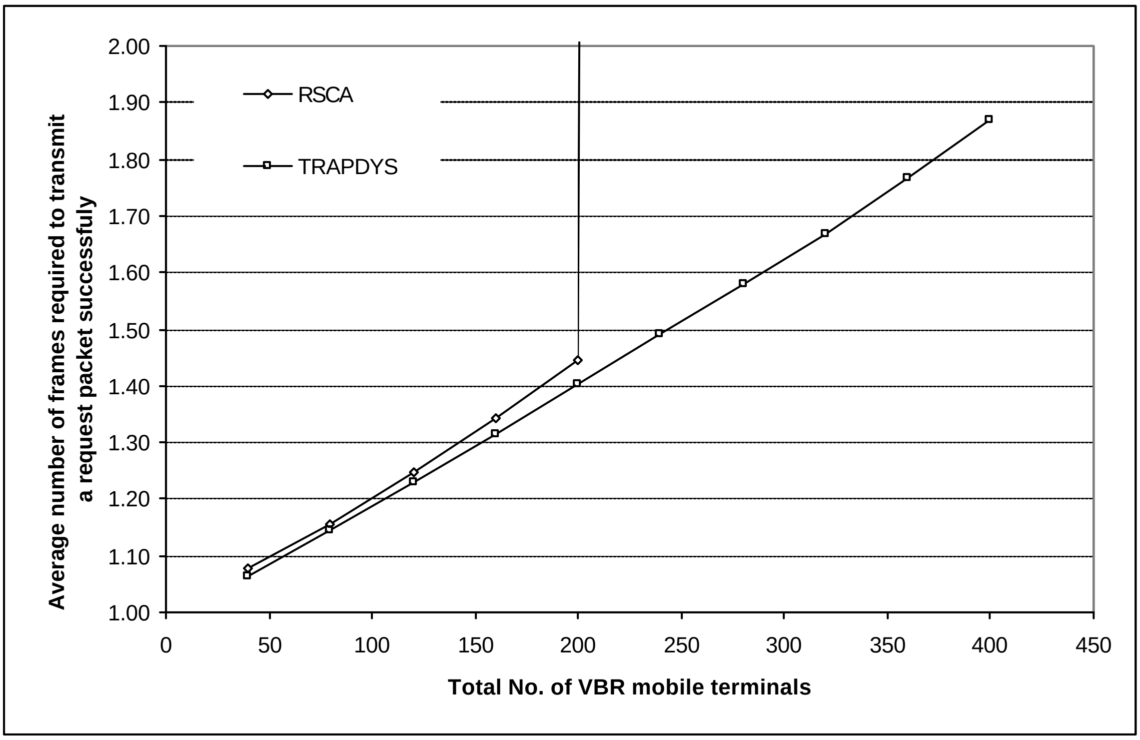

Figure 5.12: Average delay of the four classes of RSCA and TRAPDYS

Case two

Here, the results for RSCA are the same as in the first case study, except for Class 3. The number of MTs in Class 3 is constant and therefore the average delay is the same throughout our investigation. Because the classes under RSCA do not interfere with each other, their mean delays do not change. The results for TRAPDYS are very different. Figure 5.13 shows the mean delay experienced by Class 1 and Class 2 under TRAPDYS and RSCA. Class 1 under TRAPDYS has a lower delay than under RSCA. The break-off point that is observed at 50 MTs in RSCA is not observed in TRAPDYS. In Class 2, the performance under TRAPDYS is not better than under RSCA for the number of terminals smaller than 60 MTs. Under RSCA, the mean delay has a steeper gradient than under TRAPDYS. No break off point is observed in TRAPDYS.

Figure 5.14 depicts results for Class 3 and Class 4, for the two protocols. In Class 4, the curve plotted for TRAPDYS is nearly parallel to the curve plotted for RSCA and has a large delay. Usage of request slots by Class 4 is high even when the number of MTs assigned to this class is low. We focus on the performance of Class 3 in this study. TRAPDYS has decreased the delay of Class 3 by a large margin when the number of MTs in other classes is low. The traffic has been distributed over other classes. As the number of MTs increases, the delay experienced under TRAPDYS begins to approach the delay experienced under RSCA. The two curves merge at 80 MTs. The TRAPDYS protocol reduces the traffic for slots of Class 3 by utilizing the request slots of other classes. The overall performance of TRAPDYS and RSCA is shown in Figure 5.15. The lines plotted for TRAPDYS have a lower gradient than the lines plotted for RSCA in all classes. The delays of RSCA rise faster than TRAPDYS. This indicates that TRAPDYS is more stable than RSCA.

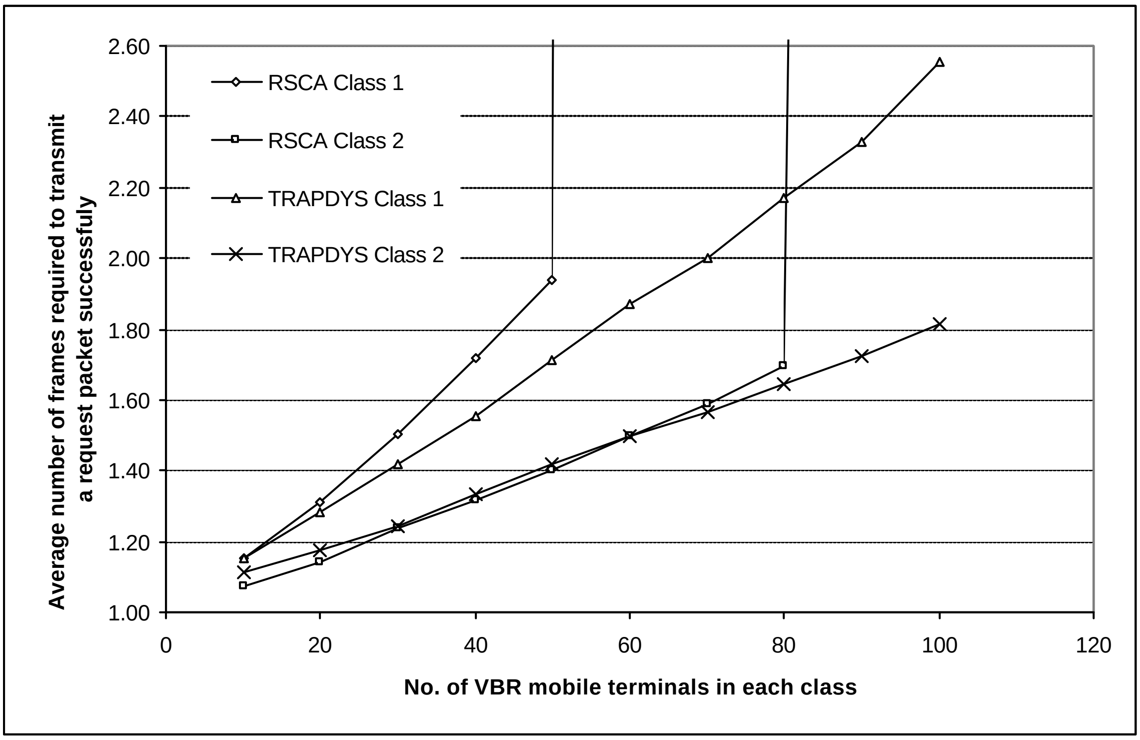

Figure 5.13: Average delay of Class 1 and Class 2 of RSCA and TRAPDYS with heavy traffic flow in Class 3.

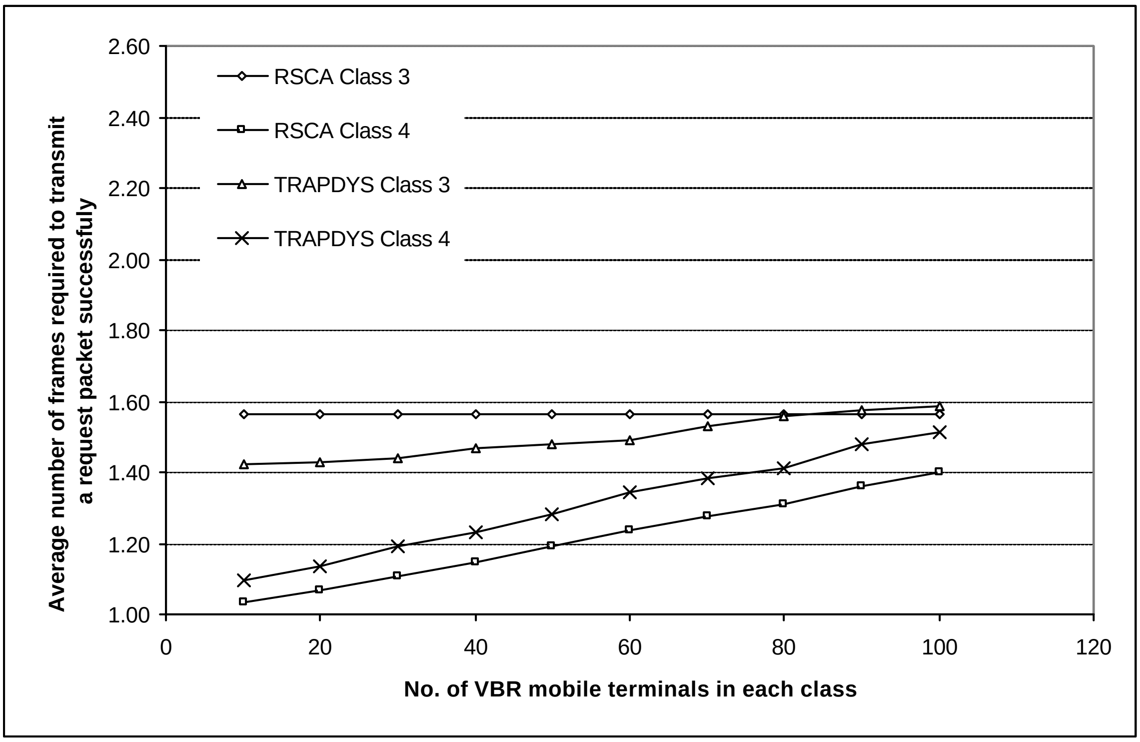

Figure 5.14: Average delay of Class 3 and Class 4 of RSCA and TRAPDYS with heavy traffic flow in Class 3.

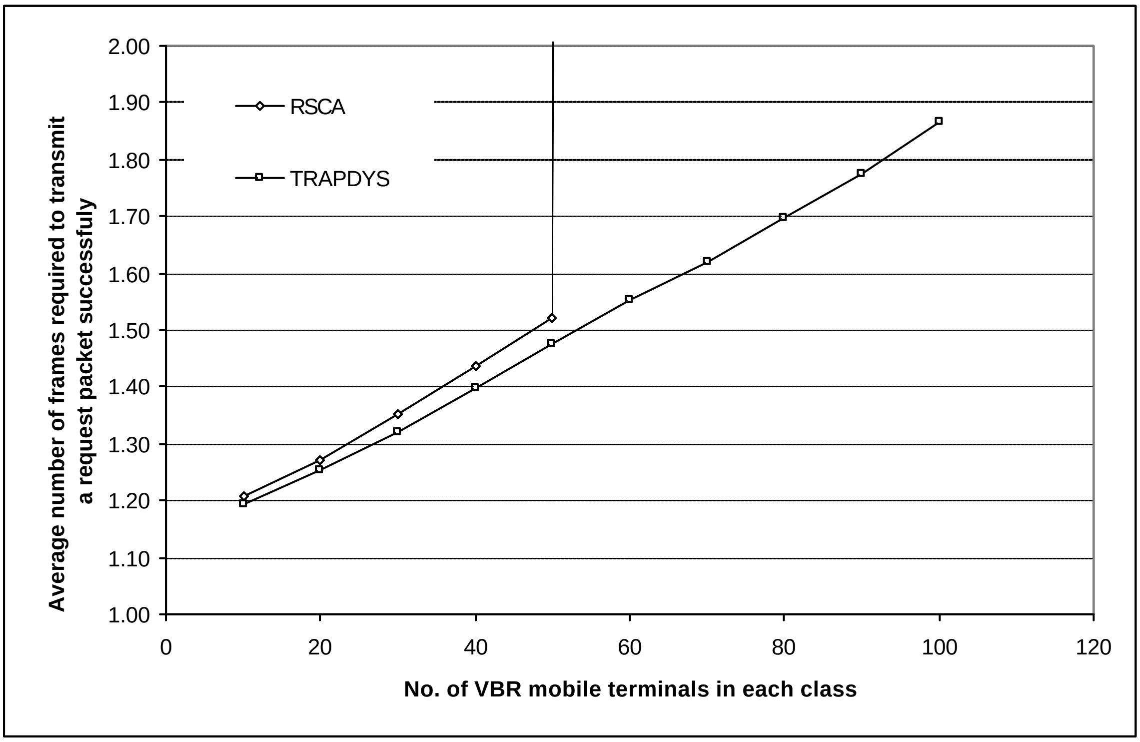

Figure 5.15: Average delay of the four classes of RSCA and TRAPDYS with heavy traffic flow in Class 3.

5.2.4 Study 4: The Behaviour of TRAPDYS under Burst of Requests

A traffic burst can introduce a sudden increase of the number of requests for transmission in a particular class. Such events could occur when a base station recovers from its failure or when the number of active MTs suddenly increases. A well designed MAC protocol is required to deal with such events. In this study, we focus on the behaviour of TRAPDYS and RSCA when such bursts occur. We define a burst as a number of request packets that need to be simultaneously transmitted (a packet from each MT). Terminals begin their attempts of transmitting of request packet during the same frame. Once their requests have been accepted, they become inactive again.

The class and request slot layout use in this study is same as previously. The TRAPDYS parameters are the same as Study 3. All classes contain zero MT before a burst is introduced. This will provide a fair environment for comparing RSCA and TRAPDYS. If steady state traffic is used, then TRAPDYS is likely to outperform RSCA before a burst appears (see results in Study 3). In our first experiment, we look at the effect of a burst of request generated by 20 MTs in Class 1. Then we observe the behaviours of the two protocols under a burst of requests caused by 25 MTs. We also look into the relationship between the number of collected observation from simulation and the relative error of the final results.

Since TRAPDYS is a protocol based on prioritisation, it is required to provide different levels of priority between classes even when a burst of requests occurs. To observe such behaviour, we introduce a burst in Class 1 and a burst in Class 3 at the same time. The two bursts are equal in size, and both contain 30 requests. A burst in Class 3 is expected to subside faster than the burst in Class 1.

In the first experiment, we introduce a burst of requests for transmission from 20 MTs in Class 1. We simulate each protocol three times with different pseudo random numbers. Figure 5.16 shows the number of MTs left to be served over time after a burst occurs. The thin lines represent the results from the three simulation replications of each protocol. The thick lines are the average over the three thin lines. Once can see that, at Frame 0, there are 20 MTs to be served by the BS. All MTs begin their transmissions of their requests at Frame 1. When a curve reaches zero, the BS finishes serving all MTs introduced by the burst. In the figure, TRAPDYS serves the MTs faster than RSCA and offers shorter delays. Its average serving time for a burst coming from 20 MTs is 42 frames. The number of MTs to be served decreases in a steady fashion with a sharp gradient. The first three quarter of the plot is nearly a straight line. The service speed decreases when the plot reaches the last quarter. This is because the number of MTs is small and their transmissions are dependent on the collision resolution scheme used, in this case on Slotted-Aloha. RSCA requires 76 frames to serve all the bursting MTs. Thus, one can see that TRAPDYS is more effective in relieving a traffic burst than RSCA.

Figure 5.17 shows a scatter point plot of the results of a single simulation replication of RSCA. Each point represents the number of transmissions occurring in Class 1 in time. If the number of transmissions drops to one, then the transmission is a successful transmission. If the number of transmissions is greater than one, then a collision has occurred. By counting the number of frames between the frame when the burst is introduced and the frame where the transmission becomes successful, one can find out the delay of a given request packet. For example, a burst occurs at Frame 6 and the first successful transmission of a request occurs at Frame 15, thus the delay of transmitting that request is 10 (15-5). The figure shows a collision of 20 requests from MTs in Frame 6. This is the point where the burst is introduced. All 20 MTs become active and transmit at the same frame. Since there is only one request slot in Class 1, the 20 request packets from the MTs collide. The last packet is successfully transmitted at Frame 81 with a delay of 76 frames (81-5).

Figure 5.18 shows a scatter point plot for TRAPDYS under the same assumption. Again, this is from a single simulation. TRAPDYS moves a portion of traffic in Class 1 to slots of classes that do not have any traffic. Points of different shapes show the request slots that have been used in transmissions. When a burst from 20 MTs is introduced, only 15 MTs transmit their request packets in slots of Class 1. The other five MTs have transmitted in slots for the other classes. The chance of a successful transmission is great since slots of the other classes have low usage. The last successful transmission occurs in Frame 48 with a delay of 43 frames. Only ten of the twenty MTs have transmitted their request packet through Class 1.

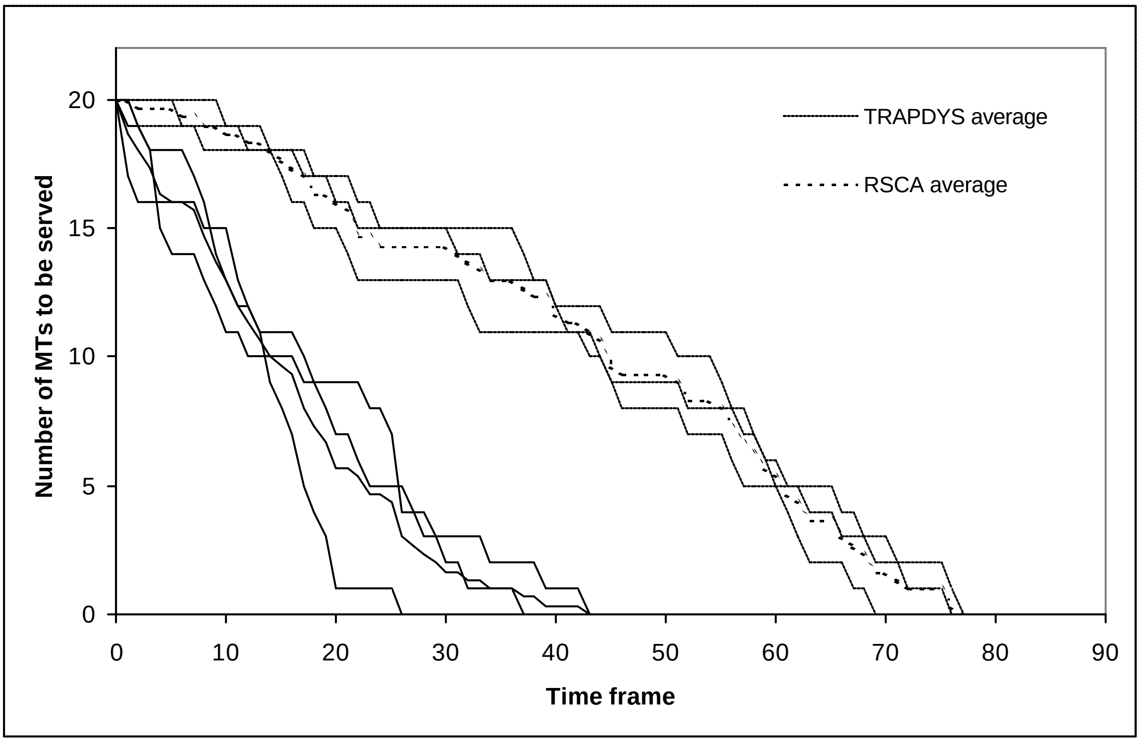

Figure 5.16: Number of MTs to be served over time of RSCA and TRAPDYS after a burst of request from 20 MTs vs. time.

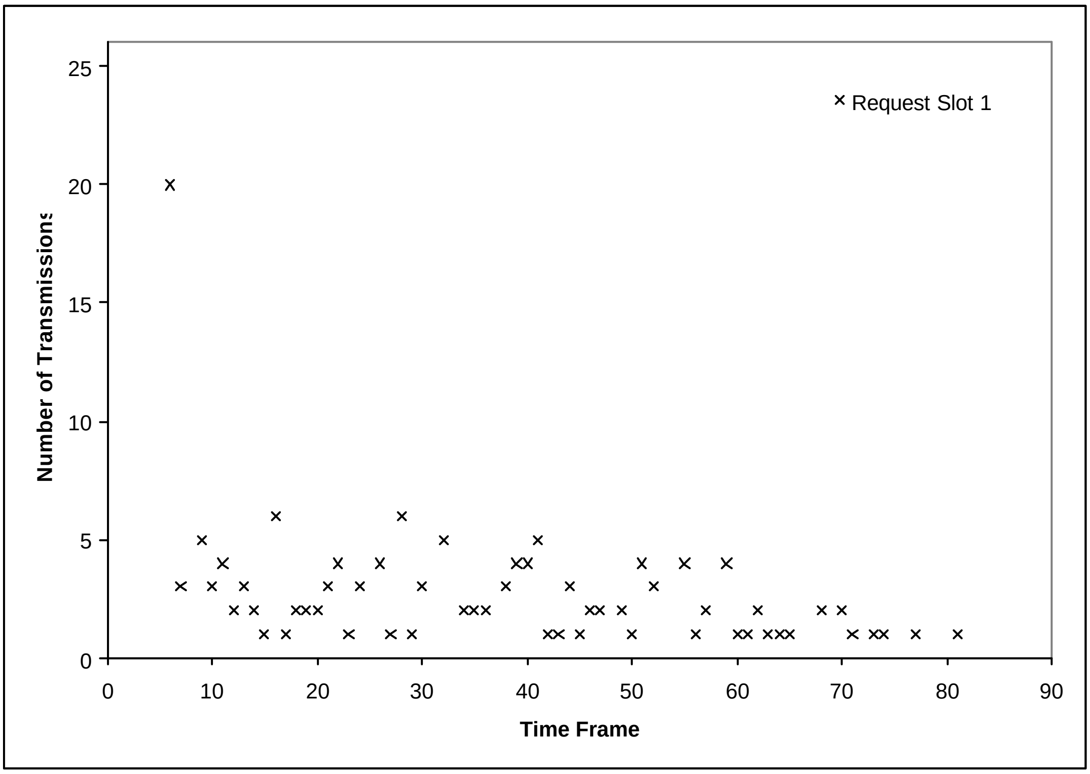

Figure 5.17: Number of transmissions of RSCA after a burst of requests from 20 MTs vs. time.

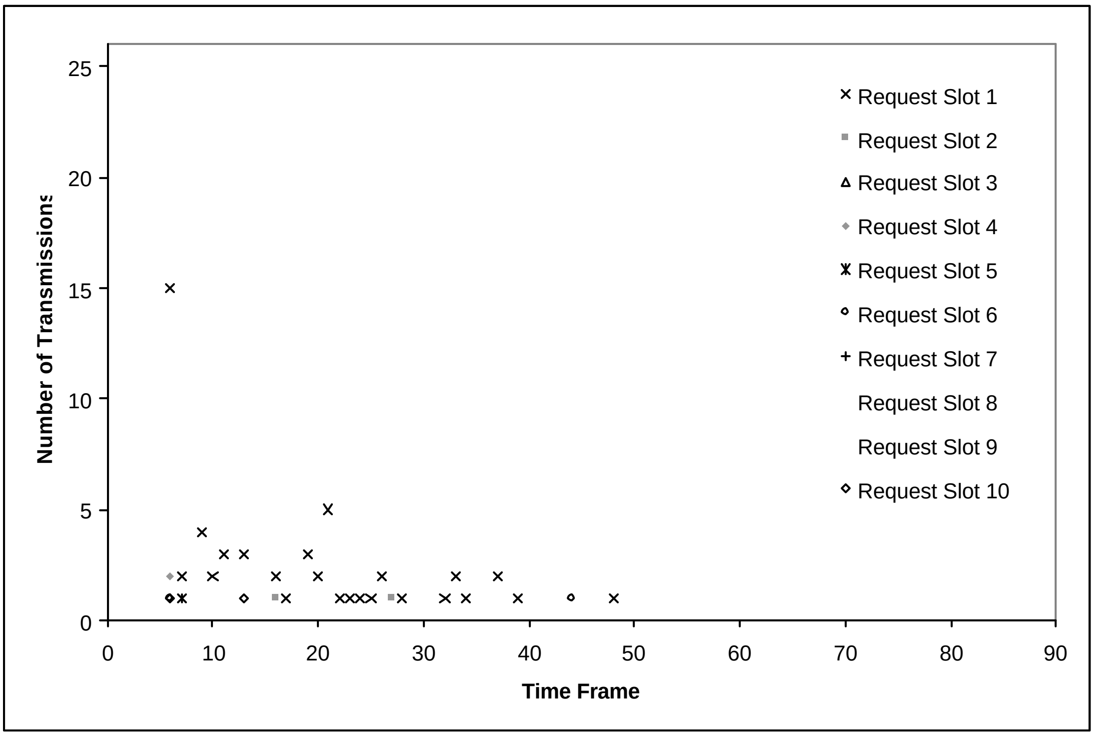

Figure 5.18: Number of transmissions of TRAPDYS after a burst of requests from 20 MTs vs. time.

Figure 5.19 shows the number of MTs to be served over time under RSCA and TRAPDYS when a burst of request from 25 MTs is introduced in Class 1. One can see that TRAPDYS serves MTs faster than RSCA. RSCA requires 129 frames to serve 25 MTs. Since the MTs can only transmit in one request slot, a large number of collisions occur. The plot for TRAPDYS has a sharper slope. The average serving time of 25 requests is 54 frames. The difference of the two protocols in delay becomes greater when the size of the burst increases.

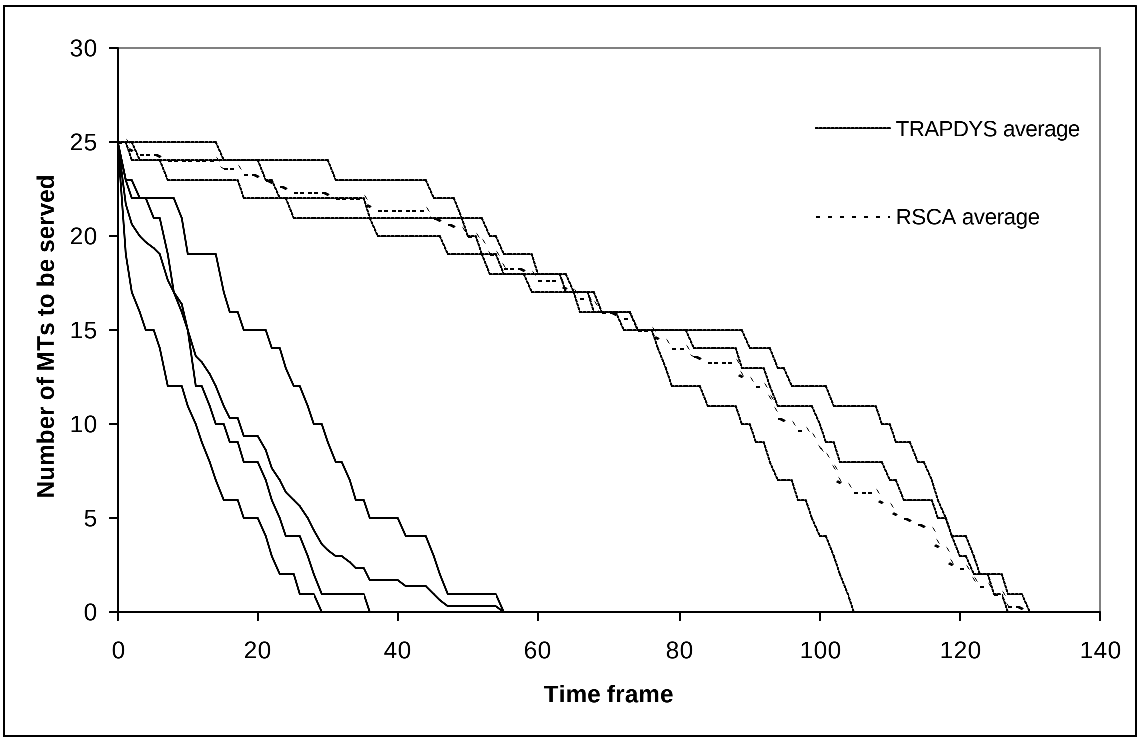

Figure 5.19: Number of MTs to be served over time of RSCA and TRAPDYS after a burst of request from 25 MTs vs. time.

The level of precision in the represented result is an important issue. Results obtained from small numbers of simulation replications can be unreliable. Figure 5.20 shows the results obtained for RSCA for a burst of request from 20 MTs for different numbers of simulation replications. Curves become smoother when the sample size increases. In the previous cases (Figures 5.16 and 5.19), we have chosen to take three replications. The relative errors of each averaged point reflect the level of credibility of the results. Figure 5.21 shows relative errors of results presented in Figure 5.20. Relative errors are produced at 95 percent confidence level. One can see that errors increase greatly as the curves pass a point where only one MT is left to be served (Frame 75 for 3 replications, Frame 83 for 100 replications, Frame 82 for 1000 replications, and Frame 83 for 4000 replications). This is the region under the grey horizontal line in Figure 5.20. The occurrence of a transmission after this point of time is very random and rare. A large number of replications are required to collect a satisfactory large sample. Due to these reasons, we exclude this region from our discussions. Such regions are with too large error to be credible, and thus should not be used for drawing any conclusions. Figure 5.22 shows the same plots as in Figure 5.21 but without this region. The figure shows that plots with large number of samples (replications) have lower relative errors. The highest error, in the case of three replications, is 113% and the highest error for four thousand replications is 7%. If one requires results with less than 10% errors, four thousand replications should be executed.

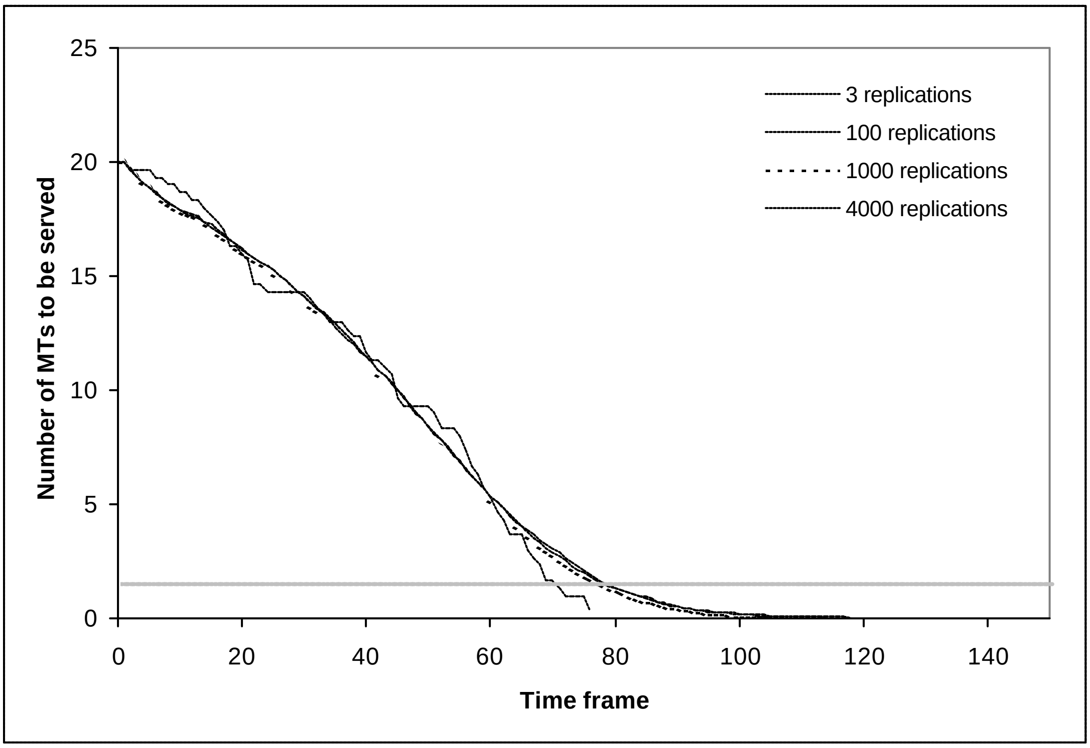

Figure 5.20: Averaged results for RSCA with different numbers of simulations replication after a burst of request from 20 MTs vs. time.

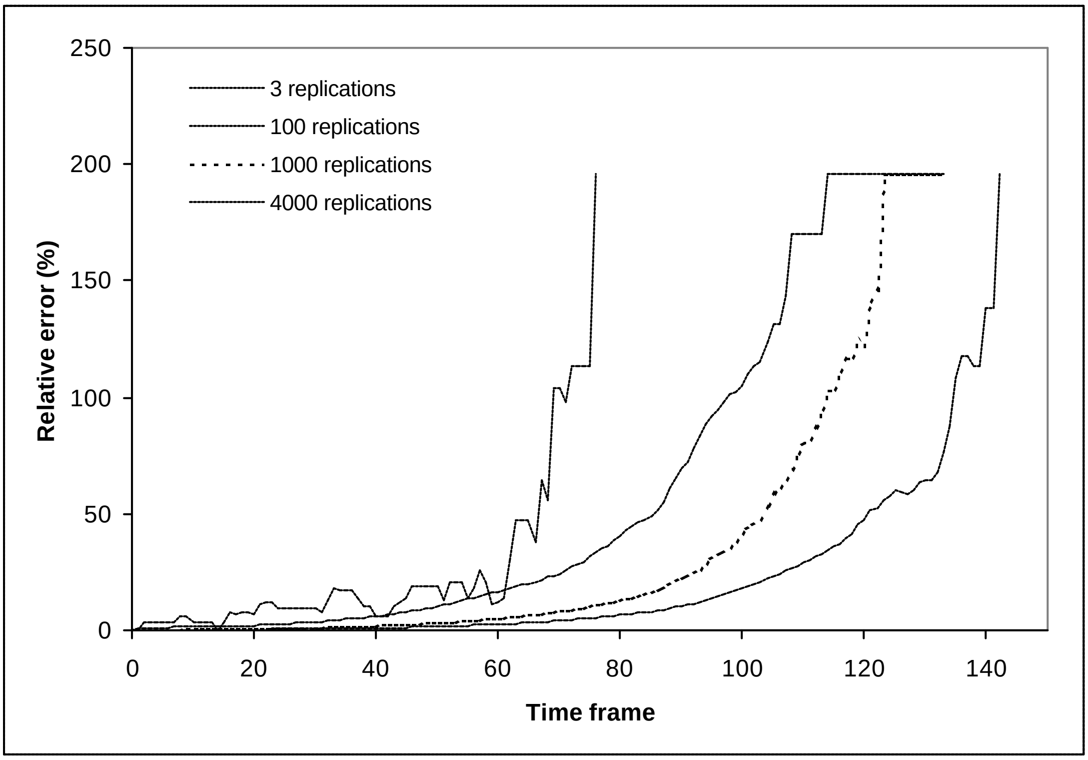

Figure 5.21: Relative errors of the averaged results for RSCA after a burst of request from 20 MTs vs. time.

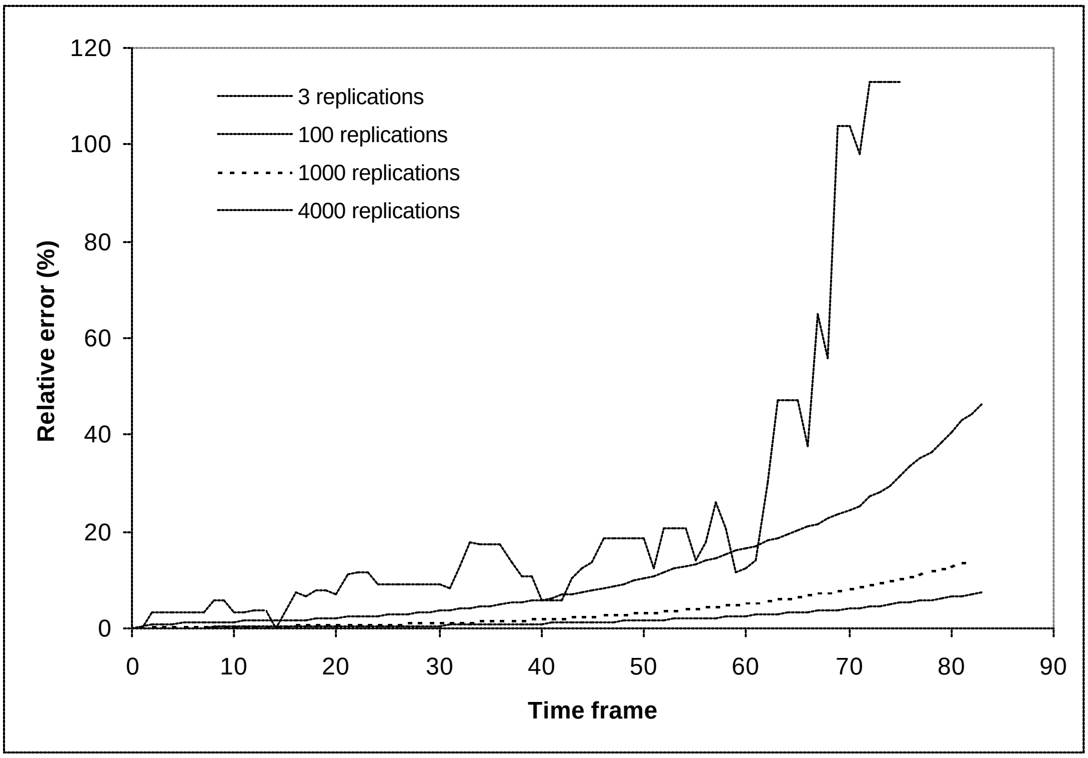

Figure 5.22: Relative errors of the averaged results for RSCA with out a tail after a burst request from of 20 MTs vs time.

The results for the study of maintainability of class priority under TRPADYS are presented in Figure 5.23. It shows the number of MTs to be served by the BS of Classes 1 and 3 versus time, when one burst occurs in Class 1 and another in Class 3. Both curves are relatively straight in the first three quarters. The average time required to serve 30 requests of Class 1 is 83 frames. The time required to serve 30 requests of Class 3 is 35 frames. Thus, the request packets of Class 3 are served much faster than that of Class 1 from the very beginning. Faster service results in shorter delays and effectively provides different priorities in different classes. The figure shows that TRAPDYS can still provide prioritised service when bursts of traffic occur.

Figures 5.24 and 5.25 are the scatter point graphs obtained from single simulation replications for transmission of Class 1 and Class 3. The MTs of both classes use request slots outside the slots of their own class to transmit their request packets. Most Class 1 transmission occur in Request Slot 1 and most Class 3 transmissions occur in Request Slots 4, 5, and 6.

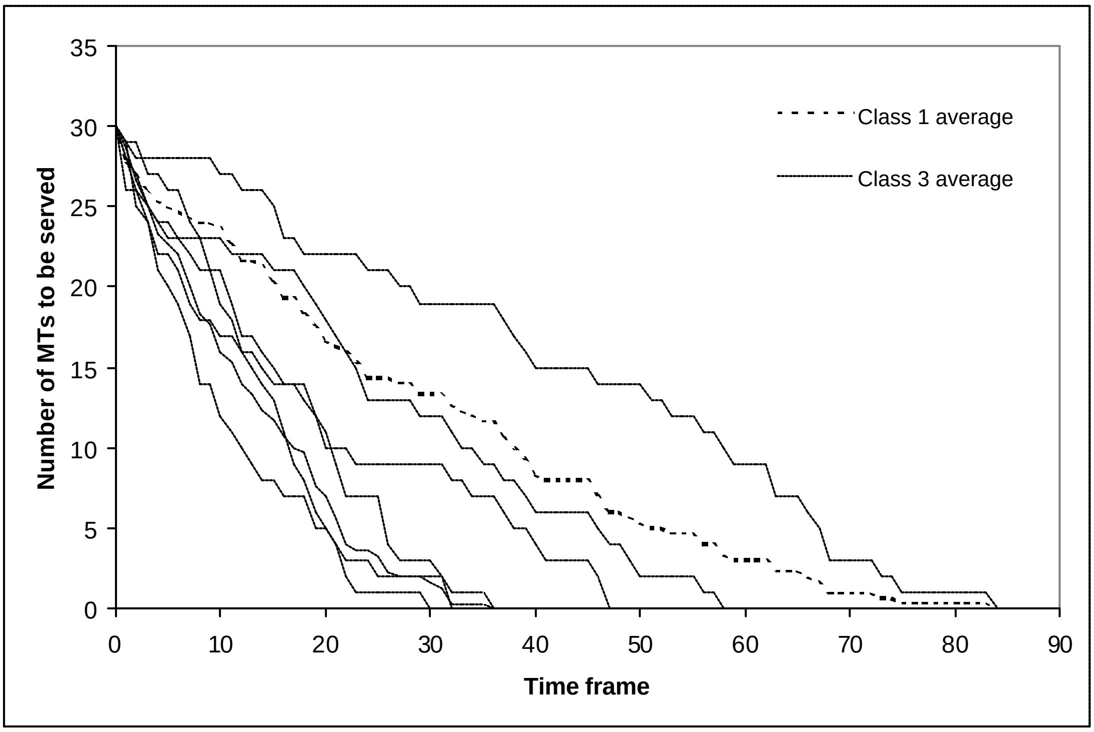

Figure 5.23: Average number of MTs to be served over time in Class 1 and Class 3 using TRAPDYS when two bursts of request are introduced.

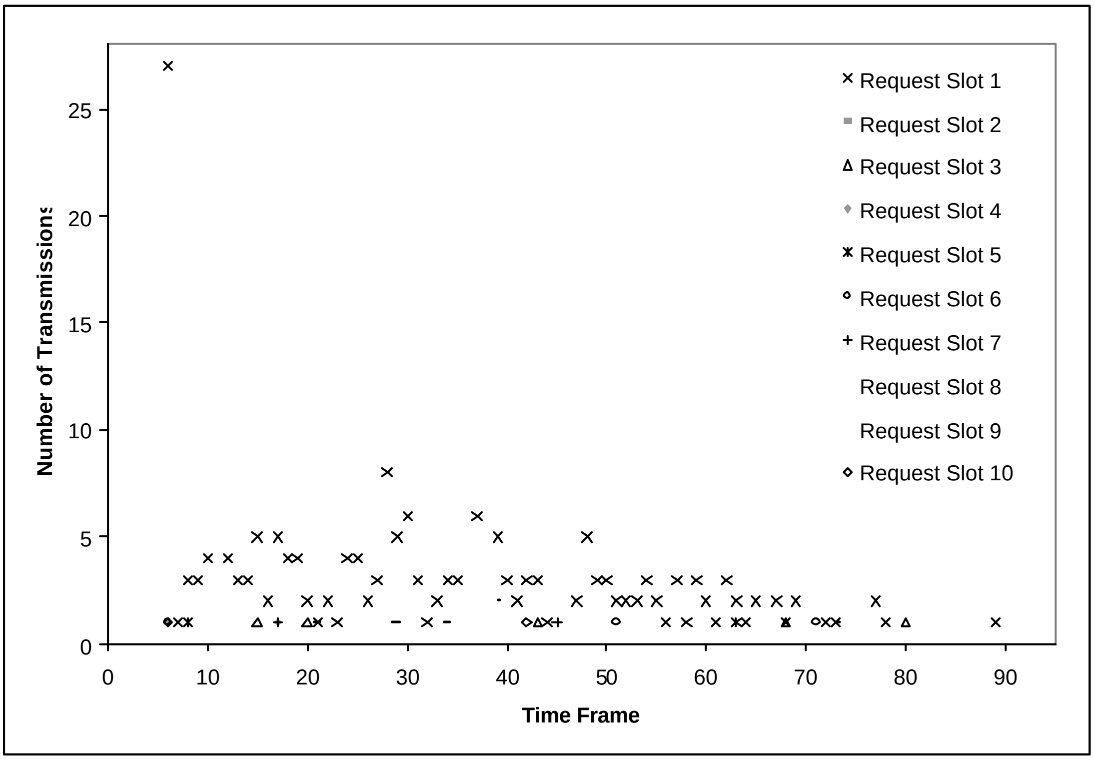

Figure 5.24: Number of transmissions of request from Class 1 traffic using TRAPDYS vs. time.

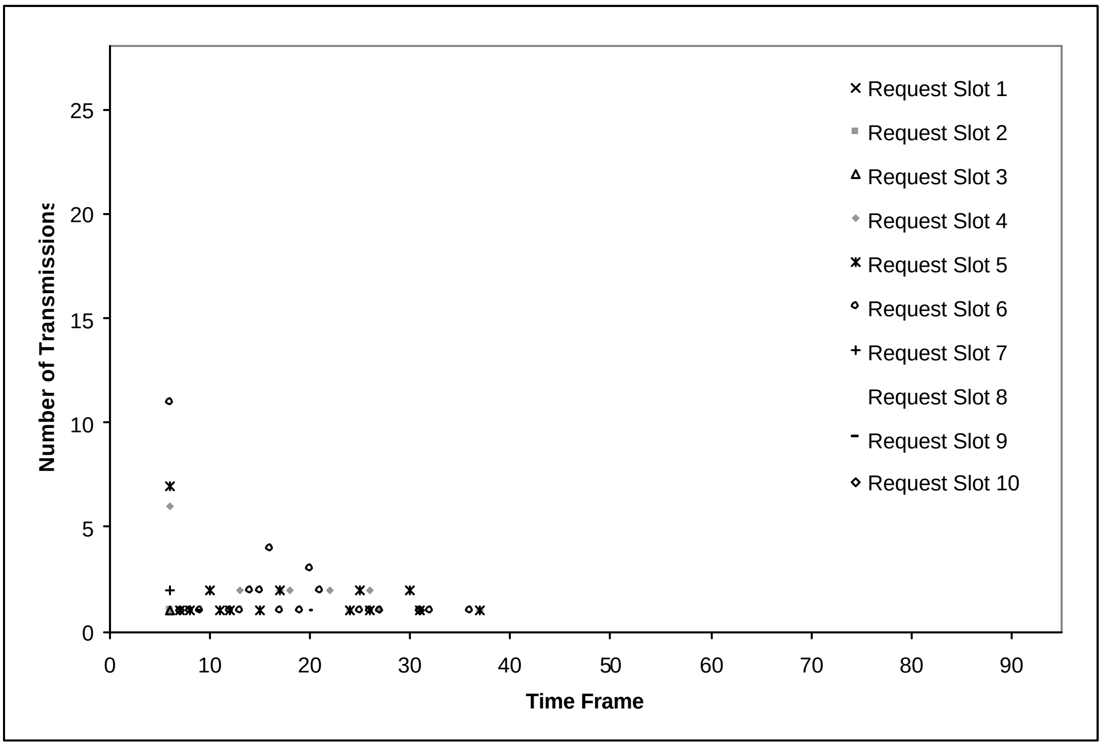

Figure 5.25: Number of transmissions of request from Class 3 traffic using TRAPDYS vs. time.

5.2.5 Study 5: Effects of Different Numbers of Classes and Request Slots

Now, let us investigate the behaviour of TRAPDYS when the number of classes of mobile terminal and the number of request slots changes. Under different numbers of classes and request slots in the reservation phase, the TRAPDYS protocol is expected to remain stable and perform similar to that of Study 3 discussed above. Here we focus on two priority classes: Class 1 and Class 2. The number of request slots for Class 1 is fixed to one in all scenarios. The number of request slots for Class 2 increases one by one, from one to four. Four scenarios have been designed (see Figure 5.26).

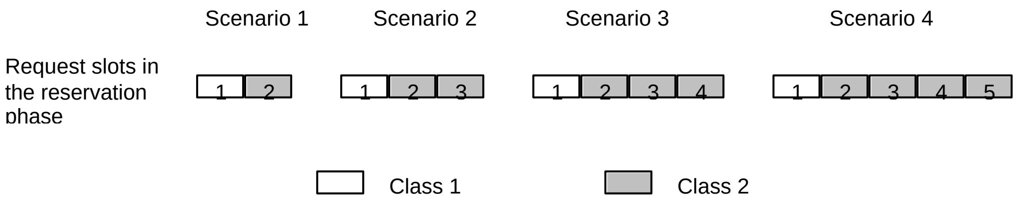

Figure 5.26: Scenarios use for studying the effects of numbers of classes and request slots.

When the number of request slots is equal to one, the two classes have equal priority. When the number of request slots is greater than one, Class 2 has higher priority than Class 1.

Figure 5.27 shows the results when the number of request slots equals one in both Class 1 and Class 2. The quality of service experienced by each of the two classes is identical. The mean delays for the two classes have a break-off point at 50 MTs. This is expected since the two classes have the same number of request slots and the same traffic density.

Figure 5.28 shows the results of Class 1 with one request slot and Class 2 with two request slots. A break-off point in Class 1 occurs when number of MTs reaches 90. In comparison to the first scenario, the maximum capacity of Class 1 has been increased by nearly two fold. A clear difference is observed though, since Class 2 experiences lower delay than Class 1. Service of Class 2 terminal when reaching the break-off point is somewhat different from that of Class 1. The delay rises sharply before a break-off point occurs.

Figures 5.29 and 5.30 show the results obtained for Scenarios 3 and 4. The difference in average delay experienced by Class 1 and Class 2 becomes greater when the number of request slots assigned to Class 2 becomes greater. The maximum capacity of Class 1 increases when the number of Class 2 request slots increases. The performance offered for Class 1 terminals has been improved by slightly sacrificing the performance offered to terminals of Class 2. This effect becomes less noticeable as the number of request slots assigned to Class 2 increases.

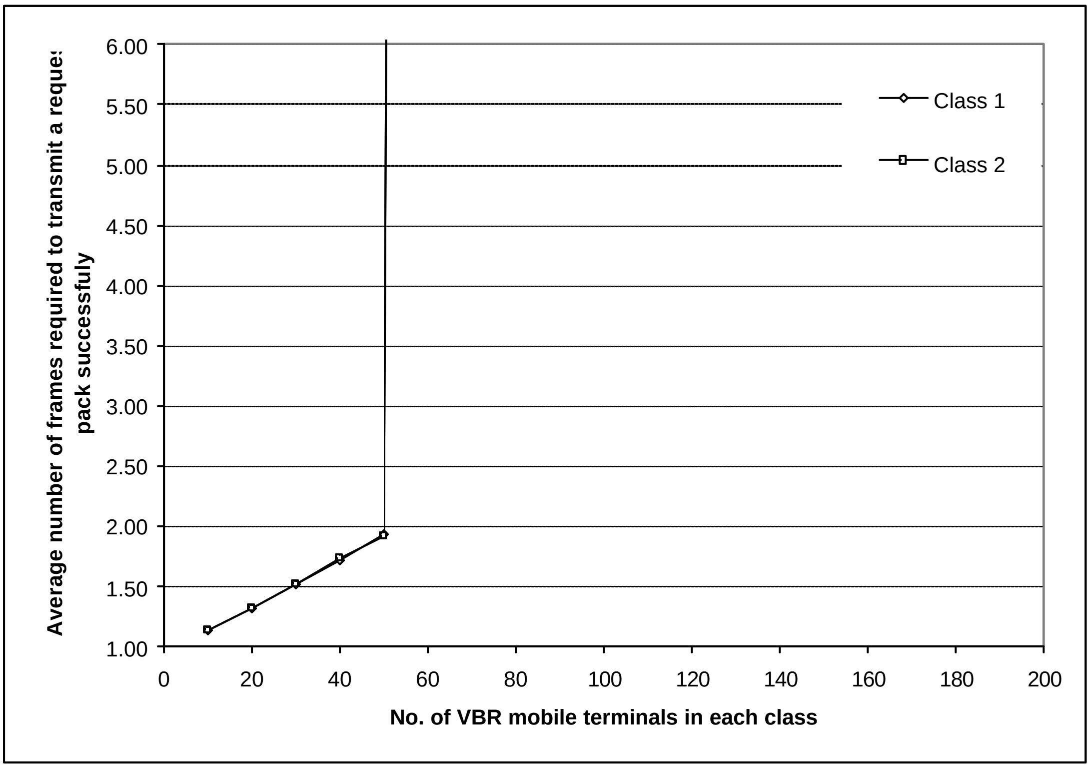

Figure 5.27: Average delay of request from Class 1 and Class 2 in Scenario 1.

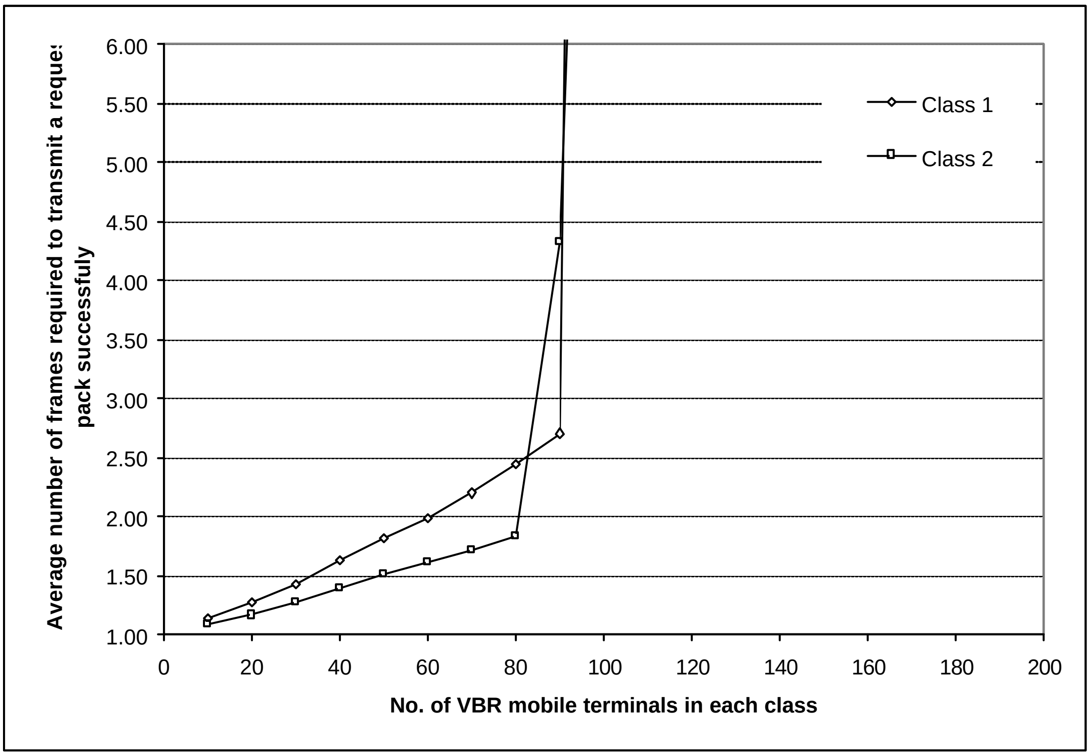

Figure 5.28: Average delay of request from Class 1 and Class 2 in Scenario 2.

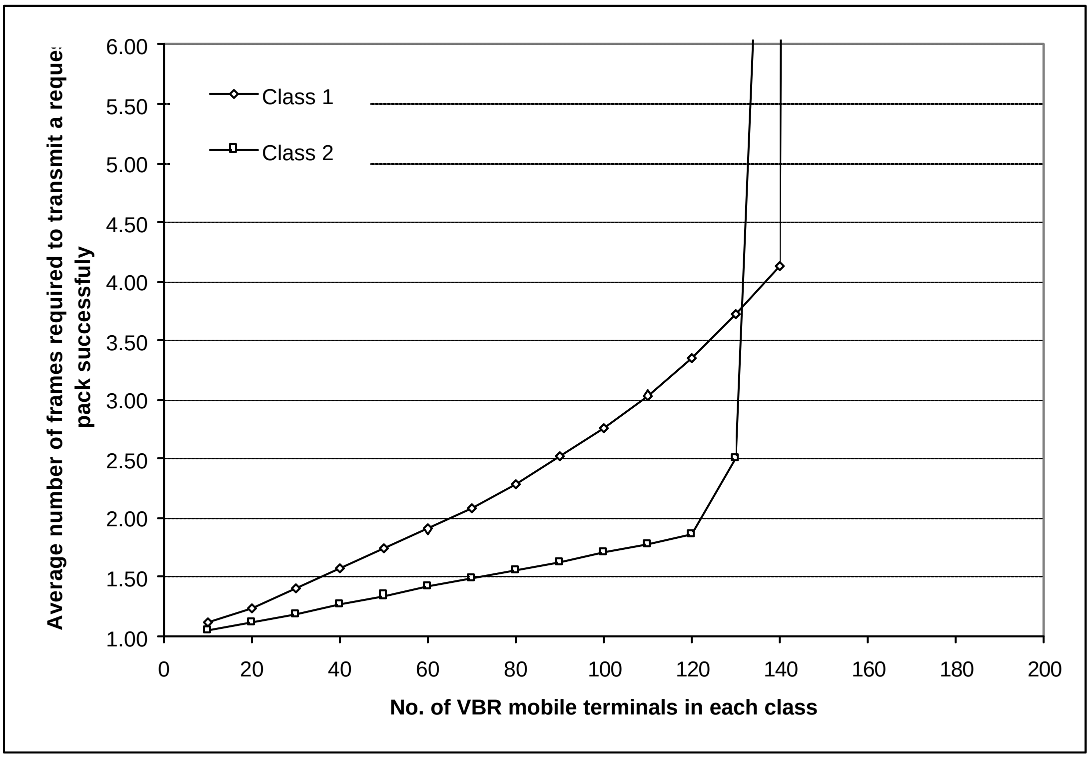

Figure 5.29: Average delay of request from Class 1 and Class 2 in Scenario 3.

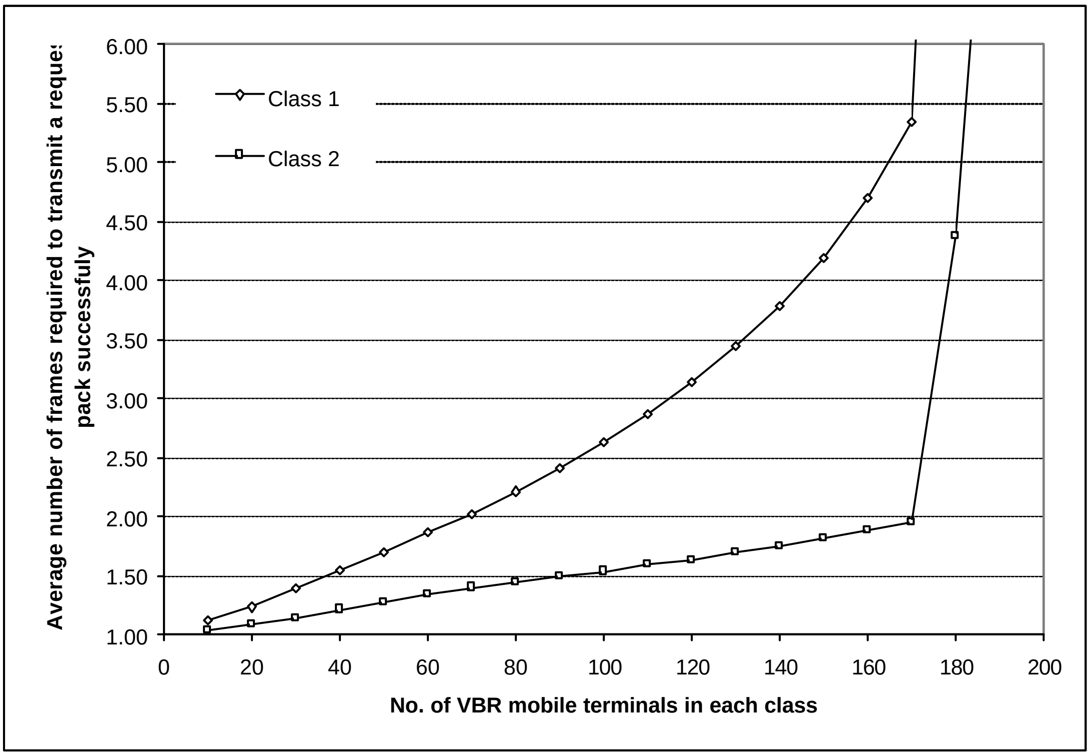

Figure 5.30: Average delay of requests from Class 1 and Class 2 in Scenario 4.

5.3 Summary

In this chapter, we have evaluated performance of TRAPDYS and compared it with RSCA. Performance evaluation was done on the basis of requests obtained from stochastic simulation. Three simulation studies were designed to evaluate the performance of the TRAPDYS protocol. The first study shows the importance of conducting simulation with small statistical errors, same results from a simulation can be very unreliable. The second study shows the effects of the three parameters of TRAPDYS on its performance. In the third study, comparisons are made between TRAPDYS and RSCA. The results show that TRAPDYS offers shorter delays of requests than RSCA and can relieve heavily stressed traffic classes faster than RSCA. Overall, TRAPDYS is more stable than RSCA. The fourth study shows TRAPDYS can relieve traffic bursts move effectively than RSCA. In the last study, the effects of different numbers of classes and request slots have been investigated. In summary, the TRAPDYS protocol appears to be a very efficient solution to be used in the reservation phase of demand assignment protocols of multimedia wireless networks.

Chapter 6

Conclusions

The age of digital communication allows us to communicate with one another from anywhere in the world. With such expectations, wireless networks offer very attractive solutions necessary. Wireless devices use light waves or radio waves to transmit their signals. In order to effectively utilize the bandwidth provided by the medium, a control on its access is essential. Medium access control (MAC) is an important part of a wireless network, and many such protocols have been designed.

In Chapter 2, we have presented many issues that must be addressed when designing a wireless MAC protocol. Problems are generated by the wireless environment when signal disruptions and interferences occur frequently. Location dependent effects are generated by the position of the wireless devices. Careful design is required to overcome such problems. Protocol performance issues must also be taken into account when designing a wireless MAC protocol. The increasing popularity of multimedia applications makes quality of service (QoS) an important part of MAC protocol design. A wireless MAC protocol designed for carrying multimedia traffic should provide features that can provide a good QoS.

Wireless MAC protocols that exist today can be classified into three major classes according to their network topology: ad-hoc MAC protocols, centralized MAC protocols, and ad-hoc centralized MAC protocols. The ad-hoc MAC protocols use an ad-hoc topology in which each device in the network has the same functionality and is free to move around. The centralized MAC protocols use a centralized topology in which a base station (BS) sits stationary in the middle of the cell and organizes the transmissions between the mobile terminals (MT). The ad-hoc centralized MAC protocols combine the two. In Chapter 3, we surveyed many wireless MAC protocols of these three classes.

Among the centralized MAC protocols one called the demand assignment MAC protocol has generated interest. The demand assignment MAC protocols assign bandwidth according to the needs of the MTs. Since the bandwidth is assigned by the BS, it can be effectively utilized with little waste. The MTs request bandwidth through the request slots in the reservation phase using random access. This makes the demand assignment MAC protocols scalable and suitable for supporting a large number of MTs. In Chapter 4, we investigated some of the strategies that could further improve the performance of the demand assignment protocols. Building upon a prioritisation strategy called request slot class assignment, we introduced the concept of transmission probability assignment. This concept allows a request slot to be assigned to many different traffic classes. Based on this concept, the transmission probability based dynamic slot assignment (TRAPDYS) protocol was developed, which allows each MT in the network to observe the traffic flow of the request slots in the reservation phase. The MTs assign some of the transmission probability according to the traffic conditions. This allows the request slots with less traffic to be utilized and relieves the request slots with a heavy traffic load.

In Chapter 5, we evaluated the TRAPDYS protocol through quantitative stochastic computer simulations. The results obtained suggest that the TRAPDYS protocol can provide priority access and at the same time relieve highly stressed traffic classes. The overall performance of the TRAPDYS protocol is better than the request slot class assignment scheme. The TRAPDYS protocol increases the capacity of the classes that have request slots. It can set back the break-off points of low priority classes. A break-off occurs when the traffic becomes too heavy and causes the delay to rise sharply. This can cause undesirably long delays. In the TRAPDYS protocol, a large margin of the delays of the low priority classes is decreased by slightly increasing the delay of the high priority classes. A study of the effect of traffic bursts shows TRAPDYS is able to relieve traffic bursts quickly while providing access priorities.

A protocol based on transmission probability assignment such as the TRAPDYS protocol has a potential of providing a flexible and intelligent random accessing request channel. Although the TRAPDYS protocol shows reasonable performance, it could be further improved by improving the traffic prediction and the method for assigning the transmission probability. This is left for further research.

Acknowledgments

I would like to thank

My supervisor Associate Professor Krzysztof Pawlikowski for his guidance and support.

My co-supervisor Associate Professor Harsha Sirisena from department of Electrical and Electronics Engineering.

References

[Abra70] N. Abramson, “The Aloha system, Another Alternative for Computer Communications”, AFIPS Conference Proceedings, 1970 Fall joint Computer Conference 37, 1970, 281-285.

[Acam97] A.S. Acampora and A.V. Krishnamurthy, “A New Adaptive MAC Layer Protocol for Wireless ATM Networks in Harsh Fading and Interference Environments”, Proceeding of ICUPC ’97, Vol.:2, 1997, 410-415.

[Alas99] M. Alasti and N. Farvardin, “D-PRMA: A Dynamic Packet Reservation Multiple Access Protocol for Wireless Communications”, Proceedings of the 2nd ACM international workshop on modeling analysis and simulation of wireless and mobile systems, 1999, 41-49.

[Amit93] N. Amitay, “Distributed Switching and Control with Fast Resource Assignment/Handoff for Personal Communications Systems”, IEEE Journal on Selected Areas in Communications, volume: 11 1993, 842-849.

[Berl87] E. R. Berlekamp, E. E. Peile, and S. P. Pope, “The Application of Error Control to Communications.” IEEE Communications Magazine 25, no. 4, April, 1987, 44-57.

[Bhar98] V. Bharghavan, “Performance Analysis of a Medium Access Protocol for Wireless Packet Networks”, IEEE Performance and Dependability Symposium '98, August 1998.

[Cape79] J. Capetenakis, "Generalized TDMA: The Multi-Accessing Tree Protocol", IEEE Transactions on Communications COM-27, no. 11, October, 1979, 1476-1484.

[Chan00] A. Chandra, V. Gummalla, and J. Limb, "Wireless Medium Access Control Protocols", IEEE Communications Surveys, second quarter, 2000.

[Chen93] K. C. Chen and C. H. Lee, "RAP – A Novel Medium Access Protocol for Wireless Data Networks", Proc. IEEE GLOECOM, 1993

[Chew99] B. Chew and A. Hac, "A Multiple Access Protocol for Wireless ATM Networks", Vehicular Technology Conference Proceedings, 1999, 1715-1719.

[Chla00] I. Chlamtac, A. Myers, V. Syrotiuk, and G. Zaruba, "An Adaptive Medium Access Control (MAC) Protocol for Reliable Broadcast in Wireless Networks", IEEE International Conference on Communications, 2000, Volume 3, 2000, 1692-1696

[Crow97] B. Crow et al., "IEEE 802.11: Wireless Local Area Networks", IEEE Commun. Mag., vol. 35, no. 9, Sept, 1997, 116-126.

[Ewin99] G. Ewing, K. Pawlikowski, and D. McNickle, "Akaroa2: Exploiting Network Computing by Distributing Stochastic Simulation", Proceedings of 13th European Simulation Multi-conference, 1999, 175-181.

[Fisc92] W. Fishcher and K. Meier-Hellstern, "The Markov-modulated Poisson Process (MMPP) cookbook", Performance Evaluation, 149-171.

[Full95] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Floor Acquisition Multiple Access (FAMA) for Packet-radio Networks", Proc. ACM SIGCOMM 95.

[Garc99] J. J. Garcia-Luna-Aceves and C. L. Fullmer, "Floor Acquistion Multiple access (FAMA) in Single-channel Wireless Networks", Mobile Networks and Applications, volume 4, 1999, 157-174.

[Colo99] G. Colombo, L. Lenzini, E. Mingozzi, B. Cornaglia, and R. Santaniello, "Performance Evaluation of PRADOS: A Scheduling Algorithm for Traffic Integration in a Wireless ATM Network", Proceedings of the fifth annual ACM/IEEE international conference on Mobile computing and networking (MobiCom'99), August 1999, pp. 143-150.

[Good87] D. J. Goodman and A. Saleh, “The Near/Far Effect in Local ALOHA Radio Communications”, IEEE Trans. Vehic. Commun., VT-36 no. 1, Feb 1987

[Good89] D.J. Goodman, R. Valenzuela, K. Gayliard, and B. Ramurthy, “Packet Reservation Multiple Access for Local Wireless Communications”, IEEE Transactions on Communications, Volume: 37, 1989, 885-890.

[Gumm00] A. Gummalla and J. Limb, “Wireless Collision Detection (WCD): Multiple Access with Receiver Initiated Feedback and Carrier Detect Signal”, IEEE International Conference on Communications, 2000, Volume 1, 2000, 397-401

[Haar00] J. Haartsen, “The Bluetooth Radio System”, IEEE Personal Communications, February, 2000, 28-36.

[Haba00] M. Habaebi and B. Ali, “A New MAC Protocol For Wireless ATM Networks”, Proceedings TENCON 2000, Volume: 3, 393 –398.

[Hain93] R.J. Haines and A.H. Aghvami, “Indoor Radio Enviroment Considerations in Selecting a Media Access Control Protocol for Wideband Radio Data Communications”, IEEE International Conference on Communications ’93, Volume:2, 1993, 990-994.

[Hamm86] J. L. Hammond and P. J.P. O’Reilly, “Performance Analysis of Local Computer Networks”, Addison-Wesley Publishing Company, 1886, 291-303.

[Jain99] S. Jain, V. Sharma, and D. Sanghi, “FAFS: a new MAC protocol for wireless ATM”, IEEE International Conference on Personal Wireless Communication, 1999, 135 –139.

[Jian98] J. Jiang and T. H. Lai, “Efficient Media Access Control Protocol for Delay Sensitive Bursty Data in Broadband Wireless Networks”, IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, volume 3, 1998, 1355-1359.

[Karn90] P. Karn, “MACA – A New Channel Access Method for Packet Radio”, ARRL/CRRL Amateur Radio 9th Computer Networking Conference, 1990, 134-140.

[Karo95] M. J. Karol, Z. Liu, and K. Y. Eng, “An Efficient Demand-assignment Multiple Access Protocol for Wireless Packet (ATM) Networks”, Wireless networks 1 (1995), 267-279.

[Klei75] L. Kleinrock and F. Tobagi, “Packet Switching in Radio Channels: Part I -- Carrier Sense Multiple Access Modes and Their Throughput-delay Characteristics”, TRANSCOM COM-23(12):1400-1416, December, 1975.

[Le99] T. Le and A. Aghvami, “A MAC Protocol for Asymmetric Multimedia Traffic With Prioritized Services in Local Wireless ATM Networks”, Vehicular Technology Conference Proceedings, 1999, 123-127.

[Lehn99] P. Lehne and M. Pettersen, “An Overview of Smart Antenna Technology for Mbile Communications Systems”, IEEE Communication Surveys, volume 2, no. 4, Fourth Quarter, 1999.

[Lenz01] L. Lenzini, M. Luise, and R. Reggiannini, “CRDA: A Collision Resolution and Dynamic Allocation MAC Protocol to Integrate Data and Voice in Wireless Networks”, IEEE Journal on Selected Areas in Communications, Volume 19, No. 6, 2001, 1153-1163.

[Mikk98] J. Mikkonen, J. Aldis, G. Awater, A. Lunn, and D. Hutchison, “The Magic WAND – Functional Overview”, IEEE Journal on Selected Areas in Communications, v16, no. 6, August, 1998, 953-972.

[Pawl02] K. Pawlikowski, J. Jeong, and R. Lee, “On Credibility of Simulation Studies of Telecommunication Networks”, IEEE Communication Magazine, January 2002.

[Petr95] D. Petras, “Medium Access Control Protocol for Wireless, Transparent ATM access”, IEEE Wireless Communication Systems Symposium, Long Island, NY, November 1995.

[Pras98] R. Prasad and T. Ojanpera, “An Overview of CDMA Evolution toward Wideband CDMA”, IEEE Communications Surveys, vol. 1, No. 1, Fourth Quarter, 1998.

[Qiu96] X. Qiu, V.O.K. Li, and J. Ju, “A Multiple Access Scheme for Multimedia Traffic in Wireless ATM”, Mobile Networks and Applications 1(1996), 259-272.

[Rahn93] M. Rahnema, “Overview of the GSM System and Protocol Architecture”, IEEE Communication Magazine, April, 1993.

[Rayc94] D. Raychaudhuri and N. D. Wilson, “ATM-based Transport Architecture for Multiservices Wireless Personal Communication Networks”, IEEE Journal on Selected Areas in Communications, 12(8) October, 1994, 1401-1414.

[Rezv99] M. Rezvan, K. Pawlikowski, and H. Sirisena, “An Adaptive Reservation Scheme for VBR Voice Traffic in Wireless ATM Networks”, The 1999 Symposium on Performance Evaluation of Computer and Telecommunication Systems, 1999.

[Robe75] L.G. Roberts, "Aloha Packet System with and without Slots and Capture," ACM SIGCOMM, Computer Communication Review, Vol 5, No. 2, April, 1975.

[Sala98] D. Sala J. Limb, and S. Khaunte, “Adaptive Control Mechanism for Cable Modems MAC Protocols”, Proceeding of INFOCOM ’98, March, 1998.

[Tang00-1] K. Tang and M. Gerla, “Random Access MAC for Efficient Broadcast Support in Ad Hoc Networks”, IEEE Wireless Communications and Networking Conference, 2000, volume: 1, 454-459.

[Tang00-2] Z. Tang and J. J. Garcia-Luna-Aceves, “Collision-Avoidance Transmission Scheduling for Ad-Hoc Networks”, IEEE International Conference on Communications, 2000, Volume 3, 2000, 1788-1794.

[Tang99] Z. Tang and J. J. Garcia-Luna-Aceves, “A Protocol for Topology-dependent Transmission Scheduling in Wireless Networks”, in Proceedings IEEE WCNC, 1999.

[Toba75] F. A. Tobagi and l. Kleinrock, “Packet Switching in Radio Channels: Part II – The Hidden Terminal Problem in Carrier Sense Multiple Access Modes and the Busy-tone Solution” IEEE Trans. Communication, COM-23, 1975, 1417-1433.

[Wong00] T. Wong, J. Mark, and K. Chua, “Access and Control in a Cellular Wireless ATM Network,” IEEE International Conference on Communications, 2000, Volume: 3, 1524 –1529.

[Wong93] W. Wong and D. Goodman, “Integrated Data and Speech Transmission Using Packet Reservation Multiple Access”, IEEE International Conference on Communications ’93, 1993, 172-176.

[Wu00] S. Wu, C. Lin, Y. Tseng, and J. Sheu, “A New Multi-channel MAC Protocol with On-demand Channel Assignment for Multi-hop Mobile Ad Hoc Networks”, International Symposium on Parallel Architectures, Algorithms and Networks, 2000, 232 –237.

[Wu88] C. Wu and V. Li, “Receiver-initiated Busy-tone Multiple Access in Packet Radio Networks”, Proceedings of the ACM Workshop on Frontiers in Computer Communications Technology, 1988, 336-342.

[Wu93] G. Wu, K. Mukumoto, and A. Fukuda, “An Integrated Voice and Data Transmission System with Idle Signal Multiple Access – Dynamic Analysis”, ICICE Trans. Communication. Vol. E76-B, no. 11, Nov., 1993.

[Wu96] G. Wu et al., “An R-ISMA Integrated Voice/Data Wireless Information System with Different Packet Generation Rates”, Proceedings of IEEE ICC ’96, vol. 3, 1263-1269.

[Zhan91] Z. Zhang and A. Acampora, “Performance of a Modified Polling Strategy for Broadband Wireless Lans in a Harsh Fading Environment”, Proceeding of IEEE GLOBECOM ’91, 1991, 1141-1146.

[Zhij00] C. Zhijun and L Mi, ”SND-MAC: An Efficient Media Access Control Method for Integrated Services in High Speed Wireless Networks”, Proceedings of International Workshops on Parallel Processing, 2000, 541 –547.